REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to the Department of Defense, Executive Service Directorate (0704-0188). Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*
01-6-2012

2. REPORT TYPE
Master's Thesis

3. DATES COVERED *(From - To)*
JAN 2012 - JUN 2012

4. TITLE AND SUBTITLE
Autonomous Adaptation and Collaboration of Unmanned Vehicles for Tracking Submerged Contacts

5a. CONTRACT NUMBER
N00244-09-G-0014

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Andrew J. Privette

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Massachusetts Institute of Technology

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943

10. SPONSOR/MONITOR'S ACRONYM(S)
NPS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
1. DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Autonomous operations are vital to future naval operations. Unmanned systems, including autonomous underwater vehicles (AUVs) and autonomous surface vehicles (ASVs), are anticipated to play a key role for critical tasks such as mine counter- measures (MCM) and anti-submarine warfare (ASW). Addressing these issues with autonomous systems poses a host of difficult research challenges, including sensing, power, acoustic communications, navigation, and autonomous decision-making.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Julie Zack
			UU	99	**19b. TELEPHONE NUMBER** *(Include area code)* (831) 656-2319

Reset

Autonomous Adaptation and Collaboration of Unmanned Vehicles for Tracking Submerged Contacts

by

Andrew J. Privette

B.S., Mathematics, United States Naval Academy, 2002

Submitted to the Department of Mechanical Engineering and the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Naval Engineer
and
Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

©Andrew J. Privette, 2012. All rights reserved.

The author hereby grants to MIT permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole or in part in any medium now known or hereafter created.

Author . . . .
Department of Mechanical Engineering and the Department of Electrical
Engineering and Computer Science
May 21, 2012

Certified by . . . .
John J. Leonard
Professor of Mechanical Engineering
Thesis Supervisor

Certified by . . . .
Michael R. Benjamin
Research Scientist, Department of Mechanical Engineering
Reader

Accepted by . . . .
Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Students
Department of Electrical Engineering and Computer Science

Accepted by . . . .
David E. Hardt
Chairman, Department Committee on Graduate Students
Department of Mechanical Engineering

Autonomous Adaptation and Collaboration of Unmanned Vehicles for Tracking Submerged Contacts

by

Andrew J. Privette

Submitted to the Department of Mechanical Engineering and the Department of
Electrical Engineering and Computer Science
on May 21, 2012, in partial fulfillment of the
requirements for the degree of
Naval Engineer
and
Master of Science in Electrical Engineering and Computer Science

Abstract

Autonomous operations are vital to future naval operations. Unmanned systems, including autonomous underwater vehicles (AUVs) and autonomous surface vehicles (ASVs), are anticipated to play a key role for critical tasks such as mine countermeasures (MCM) and anti-submarine warfare (ASW). Addressing these issues with autonomous systems poses a host of difficult research challenges, including sensing, power, acoustic communications, navigation, and autonomous decision-making.

This thesis addresses the issues of sensing and autonomy, studying the benefits of adaptive motion in overcoming partial observability of sensor observations. We focus on the challenge of target tracking with range-only measurements, relying on adaptive motion to localize and track maneuvering targets. Our primary contribution has been to develop new MOOS-IvP autonomy and state estimation modules to enable an autonomous surface vehicle to locate and track a submerged contact using range-only sensor information. These capabilities were initially tested in simulation for increasing levels of complexity of target motion, and subsequently evaluated in a field test with a Kingfisher ASV. Our results demonstrate the feasibility, in a controlled environment, to localize and track a maneuvering undersea target using range-only measurements.

Thesis Supervisor: John J. Leonard
Title: Professor of Mechanical Engineering

Reader: Michael R. Benjamin
Title: Research Scientist, Department of Mechanical Engineering

Acknowledgments

Thanks to John Leonard, Michael Benjamin, and Alon Yaari for guiding me through my thesis and educational journey. Thanks to my friends, family, and classmates that have supported me during these times. And a most special thanks to my wife and best friend, Sabrina, for her patience, care, encouragement, love, and her enduring service as the rock of our family. Another special thanks to my children Sydney, Madeline, and Elijah, who gave me a smile every day, who I learned from the most in my time at MIT, and who have inspired me each day to be the best person and father I can be.

Contents

1 Introduction **17**

1.1 Thesis Goal 17

1.2 Problem Statement 18

1.3 Background 18

1.3.1 Submerged Contact Tracking 18

1.3.2 Autonomous Surface Vehicles 19

1.3.3 Autonomy Software 20

1.4 Relevance and Potential Applications 21

1.4.1 Harbor Security 22

1.4.2 Drug Interdiction 22

1.4.3 Scientific Research 23

1.5 Research Challenges 24

1.6 Thesis Contribution 26

1.7 Thesis Overview 26

2 Particle Filter **29**

2.1 Particle Filter Overview 29

2.2 Initialization 31

2.3 Particle Prediction 32

2.4 Sensor Input 33

2.5 Determination of Weights 34

2.6 Resampling 36

2.7 Particle Filter Momentum 37

3 Software Architecture and Modules 41

3.1 MOOS 41

3.2 IvP Helm 46

3.3 Range Sensor Information 47

3.4 pParticleFilterAP 47

3.4.1 Configuration Parameters for pParticleFilterAP 48

3.4.2 MOOS Variables Posted by pParticleFilterAP 51

3.4.3 MOOS Variables Subscribed by pParticleFilterAP 51

3.5 The CutRange Behavior 52

3.6 Overall Software Organization 52

4 Target Motion Analysis 55

4.1 Observability Analysis 55

4.2 Disambiguating the Particle Filter 62

4.2.1 Knowledge of Initial State 62

4.2.2 Additional Sensor 62

4.2.3 Range Correlation 62

4.2.4 Adaptive Maneuvering 64

5 Simulations, Testing, and Analysis 69

5.1 Particle Filter Settings 69

5.1.1 Measures of Effectiveness 69

5.1.2 Number of Particles 70

5.1.3 Moving Contact Settings 72

5.2 Simulation Testing 78

5.2.1 Box Maneuver 79

5.2.2 Bowtie Maneuver 81

5.2.3 Loiter Maneuver 84

5.3 Collaborative ASV Search 86

5.4 Charles River Testing 88

5.5 Analysis 89

6 Conclusions **95**

6.1 Future Work 95

A Charles River Testing Pictures **99**

List of Figures

1-1 Close up view of a Clearpath Robotic's Kingfisher 20
1-2 M100 Kingfisher 21
1-3 Top 25 U.S. Shipping Ports 23
1-4 Narco Submarine 24

2-1 Particle Filter Flow Chart 30
2-2 Particle Generation 32
2-3 Example Probability Distribution 35
2-4 Example Gaussian Distribution 36
2-5 Shift in PDF due to Resampling 37
2-6 Momentum of Particles: Particles Before Maneuver 39
2-7 Momentum of Particles: Particles After Maneuver 40

3-1 MOOS Shoreside to Multi-Vehicle Topology 42
3-2 MOOS Shoreside Community 43
3-3 MOOS ASV Community 43
3-4 MOOS Submerged Contact Community 44
3-5 Action Selection with Interval Programming 46
3-6 Software Flow Chart 53

4-1 Relative Motion 57
4-2 Relative Motion Plot of Particles at t = 0 58
4-3 Relative Motion Plot of Particles at t = 1 59
4-4 Relative Motion Plot of Particles at t = 2 60

4-5 Relative Motion Plot of Particles at t = 3 61
4-6 Sensor Correlation 63
4-7 Range Fusion at t=0 63
4-8 Range Fusion at t=1 64
4-9 Initial Localization Maneuver through Loitering 65
4-10 Intercepting Maneuver 66
4-11 Visualization of IvPHelm for the Highest Weighted Particle 67
4-12 Visualization of IvPHelm for the Average Weighted Particle Position 67
4-13 Visualization of IvPHelm of the Combined Solutions 68

5-1 Initial Particle Separation for a Stationary Contact 70
5-2 Range Convergence for the number of particles over time 72
5-3 Moving Target at t=1 73
5-4 Moving Target at t=2 73
5-5 Moving Target at t=3 74
5-6 Moving Target at t=4 74
5-7 Moving Target at t=5 75
5-8 Course Noise 77
5-9 Speed Noise 78
5-10 MIT Sailing Pavilion 79
5-11 Box Simulation 80
5-12 Box Simulation 81
5-13 Bowtie Simulation with Reserve Particles 82
5-14 Bowtie Simulation with no Reserve Particles 83
5-15 Bowtie Simulation with Reserve Particles for Uniform and Gaussian Distribution 84
5-16 Loitering Simulation 85
5-17 Loitering Simulation 86
5-18 Collaborative ASV Search 87
5-19 Charles River Testing 88

5-20 CRLB with Variance of 5 meters 91
5-21 CRLB with Variance of 10 meters 92
5-22 CRLB with Variance of 15 meters 93
5-23 CRLB with Variance of 25 meters 94

A-1 Kingfisher at sea testing 1 99
A-2 Kingfisher at sea testing 2 100
A-3 Kingfisher at sea testing 3 100
A-4 Kingfisher interna viewl 1 101
A-5 Kingfisher internal view 2 101
A-6 Up Close Tracking 102

List of Tables

1.1 M-100 Kingfisher Technical Specifications [32] 20

5.1 Table of Variables 71
5.2 Gaussian Variance Error 75
5.3 Average Course Noise Error 76
5.4 Average Speed Noise Error 77

List of Acroynms

ADCAP	Advanced Capability
AUV	Autonomous Underwater Vehicle
ASV	Autonomous Surface Vehicle
ASW	Anti-Submarine Warfare
AV	Autonomous Vehicle
CPA	Closest Point of Approach
CRLB	Cramer-Rao Lower Bound
EKF	Extended Kalman Filter
FIM	Fisher Information Matrix
GPS	Global Positioning System
IvP	Interval Programming
MCM	Mine Countermeasures
MLE	Maximum Likelihood Estimator
MPC	Modified Polar Coordinates
MOOS	Mission Oriented Operating Suite
MOOSDB	MOOS Database
MOOS-IvP	Mission Oriented Operating Suite - Interval Programming
PDF	Probability Density Function
SOSUS	Sound Surveillance System
TMA	Target Motion Analysis
USCG	United States Coast Guard
USV	Unmanned Surface Vehicle

Chapter 1

Introduction

1.1 Thesis Goal

Autonomous operations are vital to future United States naval operations. Unmanned systems, including autonomous underwater vehicles (AUVs) and autonomous surface vehicles (ASVs), are envisioned as playing a key role in critical tasks such as mine counter-measures (MCM) and anti-submarine warfare (ASW). Addressing these issues with autonomous systems poses a host of difficult research challenges, including sensing, power, acoustic communications [2, 20], navigation [18, 21, 22], and autonomous decision-making [10].

In this thesis, we address the issues of sensing and autonomy, studying the benefits of adaptive motion in overcoming partial observability of sensor observations. We focus on the challenge of target tracking with partial (range-only) information, relying on adaptive motion to localize and track maneuvering targets. Our primary contribution has been to develop new MOOS-IvP [11] autonomy and state estimation modules to enable an autonomous surface vehicle to locate and track a submerged contact using range-only sensor information. These capabilities were initially tested in simulation for increasing levels of complexity of target motion, and subsequently evaluated in a field test with a Kingfisher ASV.

1.2 Problem Statement

To confine the bounds of the problem, we made the following assumptions. This thesis focuses on tracking a single vehicle and uses a simple sonar range measurement model for the range information produced by a simulated active sonar application with at least five range measurements prior to the contact maneuvering. A single vehicle was chosen do minimize the complexity caused by both, disambiguating the contact's location and correlating the range data to the proper contact. The simulated sonar information received was in the form of direct path propagation; bottom bounce and convergence zone signal sources were not modeled. The size of the submerged contact was on the order of 10 m or less. Lastly, the setting for this thesis was in a port or harbor area (area of 1 km^2 or less), not in an open ocean environment.

1.3 Background

1.3.1 Submerged Contact Tracking

The goal for submerged contact tracking is to be able to track a submerged mobile contact given uncertain acoustic measurements of the range and/or bearing to the target. There are two basic ways for gathering information on a submerged source: passive sonar or active sonar. Passive sonar relies on the ability to listen to underwater sounds. An individual hydrophone or an array of hydrophones can perform passive sonar by receiving and processing sound waves. Examples of these are towed arrays and permanent fixtures like the sound surveillance system (SOSUS) array. Active sonar involves emitting a pulse of energy from a transmitter and sensing the reflected wave in the receiver. The range is found by measuring the time difference, Δt of the emitted and reflected waves. Because the speed of sound, v_s, in the environment is known, the range can be calculated by using the following equation: range, r, equals speed of sound multiplied by time, $r = v_s * \Delta t$.

Generally, passive sonar is used in situations in which the detector does not want to be detected and active sonar is used when that is not a concern. Active sonar can also control the energy level of the emitted pulse to overcome interfering noise from the moving detector and control the frequency at which the pulse is transmitted. Because of this, and the lack of need for stealth, simulated active sonar was used in this research.

Once the range data has been received and processed, target motion analysis (TMA) techniques are applied in concert with an estimation filter to conduct localization and perform tracking. TMA techniques are also used to overcome the problem of ambiguity that results from non-observability. Non-observability results from having a set of ranges that can produce an infinite number of course, speed, and coordinate possibilities. This topic is described in more detail in Chapter 4.

At the core of the tracking problem is the use of a recursive state estimator. Many estimators were considered, including Extended Kalman Filtering (EKF), Maximum Likelihood Estimation (MLE), and Particle Filtering (PF) [12]. For this research, a particle filter was chosen for the following reasons: (1) the ability to handle non-linear systems of equations, such as solving for coordinates and course based on range; (2) the ability to operate without distribution assumptions; (3) the capability to adapt to moving targets; (4) the production of multiple hypothesis metrics that could be directly used in MOOS-IvP, (5) its relative performance in regard to Kalman filtering and maximum likelihood estimation [31].

1.3.2 Autonomous Surface Vehicles

One of the secondary goals of this project is to develop an inexpensive, reliable, and feasible way to track and localize underwater contacts using current ASVs available at the Massachusetts Institute of Technology (MIT). MIT owns and operates two types of ASVs: the SCOUT platform, a robotic kayak that costs roughly $30,000 per unit, and a Clearpath Robotic's Kingfisher (Figure 1-1), that costs roughly $20,000 per unit. Both of these ASVs are equipped with GPS, compasses, WiFi modems, and a PC that has been configured with MOOS-IvP autonomy software. To keep costs low, and to match potential operational scenarios for active sonar tracking of submerged targets, range-only sensors were used rather than bearing-only or bearing and range.

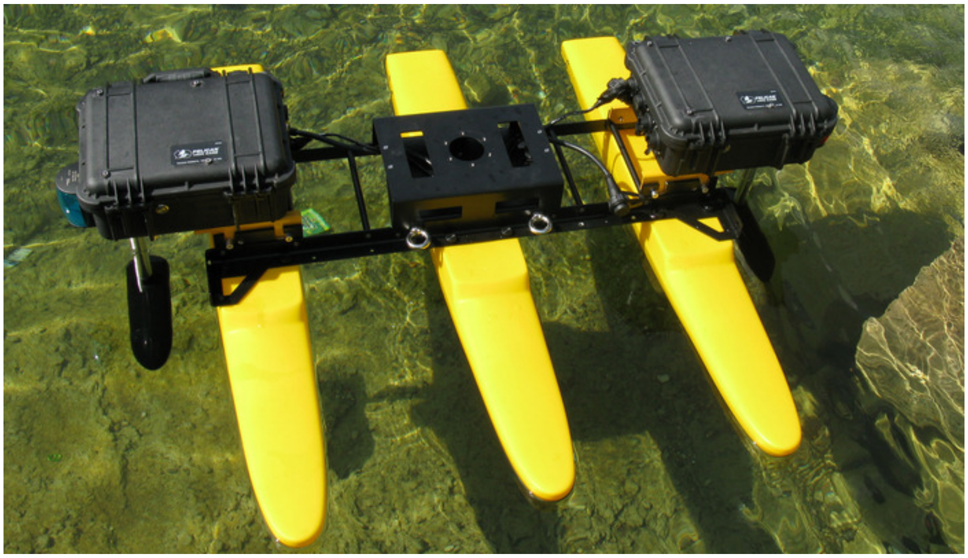


Figure 1-1: Close up view of a Clearpath Robotic's Kingfisher [32], which was used to detect and track submerged contacts. Additionally, a kingfisher was used to simulate a submerged contact [32].

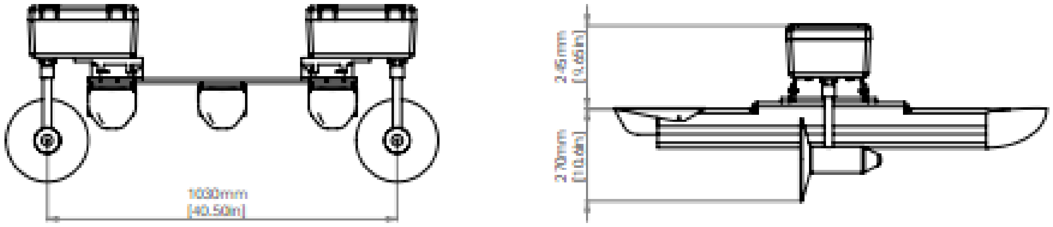
For this thesis, the M100 Kingfisher was used. Table 1.1 and Figure 1-2 contain a baseline description of the ASV. One of the benefits of using the M100 Kingfisher was the presence of two large payload boxes on top of the ASV. These boxes afford the the ASV the ability to carry additional sensors. One important restriction on the Kingfisher is that it is limited to 2 m/s maximum speed.

L x W x H	1270x1270x520mm	**Power**	200 W
Draft	280 mm	**Max Speed**	2 m/s
Weight	34 kg	**Max Thrust**	60 lb
Max Payload	15 kg	**Operating Time**	2 h
Batteries	12 V, 24 Ah	**Max Range**	10 km

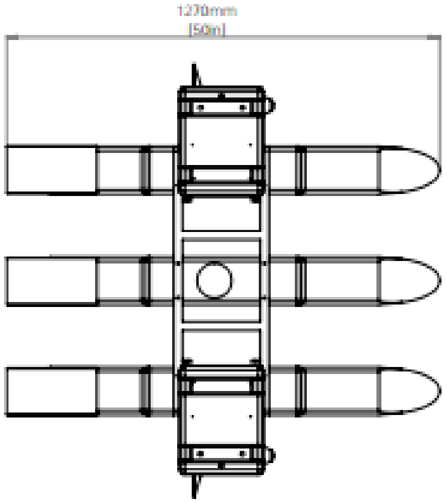
Table 1.1: M-100 Kingfisher Technical Specifications [32]

1.3.3 Autonomy Software

The autonomy software used for this thesis is a software suite known as Mission Oriented Operating Suite (MOOS), initially developed by Newman at MIT in 2001 [27].



(a) Rear View (b) Profile View



(c) Overhead View

Figure 1-2: M100 Kingfisher [32].

In 2004, Benjamin introduced the IvP-Helm, a MOOS process for mission control that that uses Interval Programming (IvP) [7] for multi-objective optimization. Many processes have been added to the MOOS structure so that this collective software suite is now known as MOOS-IvP [8]. Further documentation, missions, and software regarding MOOS-IvP can be found at www.moos-ivp.org.

1.4 Relevance and Potential Applications

The development of autonomous applications able to track submerged contacts has significant potential future applications for security, law enforcement, and scientific research.

1.4.1 Harbor Security

One of the major security weaknesses in the United States is its lack of harbor security; more specifically, the lack of security against submerged threats. The following list illustrates the vast and important role that ports and harbors have on the United States and its economy: [28]

- There are over 185 public ports in the United States.
- Over 13.3 million jobs stem from cargo-related spending.
- In 2007, there was $3.95 trillion in international trade processed in United States ports.
- In 2006, 1 billion tons of domestic goods moved via water in the United States.
- Each year, 400 million cubic yards of dredged material are removed from navigation channels in the United States.
- In 2008, $400 million was appropriated for the Department of Homeland Securitys Port Security.
- Leading commodities shipped include crude petroleum, petroleum products, chemicals, coal, and natural gas.

Most of these ports are located within the vicinity of a major metropolitan area, as may be seen in Figure 1-3. By using a method similar to the one mentioned in this thesis, harbors may be made safer from underwater threats.

1.4.2 Drug Interdiction

The United States has approximately 12,383 miles of coastline [6]; patrolling the coast for incoming drug shipments is a difficult task for the United States Coast Guard (USCG). In 2006, the authorities detected three drug-running submersibles and, later the same year, captured a submarine called *Bigfoot* and seized several tons of cocaine. In 2008, an average of ten sightings of drug-running submersibles a

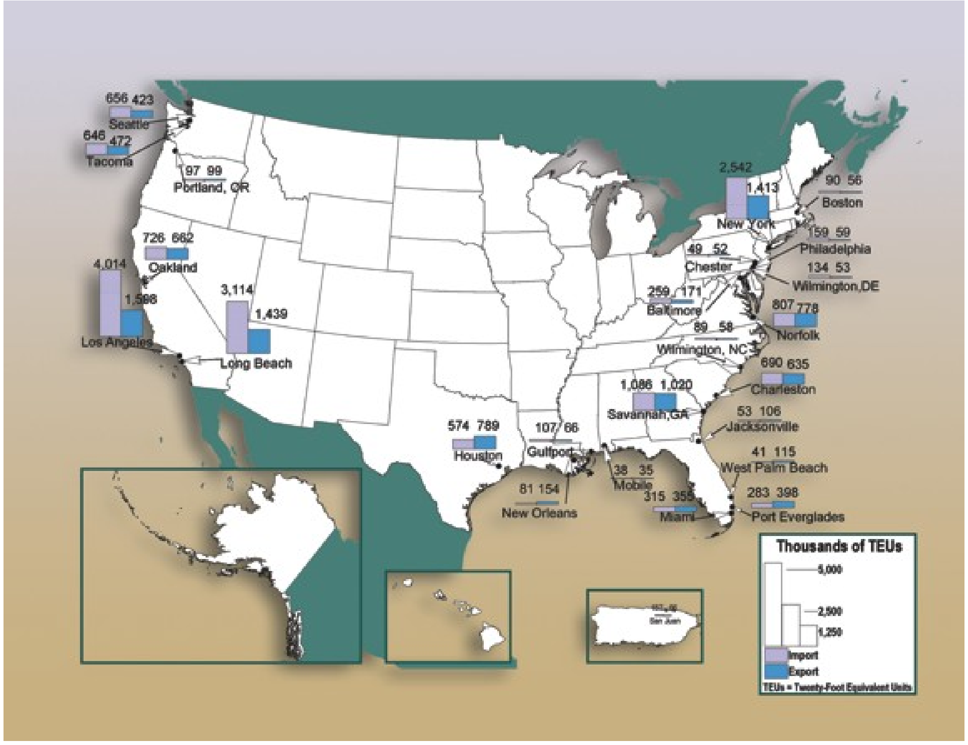

656 423
Seattle
646 472
Tacoma
97 99
Portland, OR
726 662
Oakland
4,014
1,586
Los Angeles
3,114
1,439
Long Beach
574 789
Houston
107 66
Gulfport
81 154
New Orleans
38 35
Mobile
315 355
Miami
2,542
1,413
New York
90 56
Boston
159 59
Philadelphia
49 52
Chester
134 53
Wilmington,DE
259 171
Baltimore
807 778
Norfolk
89 58
Wilmington, NC
690 635
Charleston
1,086 1,020
Savannah,GA
53 106
Jacksonville
41 115
West Palm Beach
283 398
Port Everglades
San Juan
Thousands of TEUs
5,000
2,500
1,250
Import
Export
TEUs = Twenty-Foot Equivalent Units

Figure 1-3: The top 25 U.S. Shipping Ports [29]. This maps shows the annual amount of import/export activity that takes place in the United States.

month was reported. The *narco submarine*, as seen in Figure 1-4, is the latest vehicle of choice for smuggling drugs into the United States. It has been estimated that from October 1, 2007, to February 1, 2008, that 300 tons of cocaine were smuggled into the country using a narco submarines [24].

The USCG is unable to continuously counter this smuggling threat. With the purchase and deployment of a large number of inexpensive and reliable ASVs coupled with the tracking algorithms described in this thesis, the USCG could significantly reduce the number of illegal drugs brought into the United States by means of the narco submarines.

1.4.3 Scientific Research

There are several areas of scientific research in which autonomously tracking submerged contacts can be useful One immediate need is for tracking and monitoring fish or whale populations. Autonomous tracking would provide deeper insight into

Figure 1-4: The Narco Submarine [23] was estimated to have smuggled around 300 tons of cocaine into the United States from October 1, 2007 to February 1, 2008.

sea life behaviors, migratory patterns, and population sizes. The technology is also needed to track boundaries of large masses, such as the underwater oil spill from Deepwater Horizon or the floating debris field produced by the Japanese tsunami of 2011.

1.5 Research Challenges

The problem of localizing and tracking undersea contacts raises many difficult research issues, and the literature in this field is vast. Our work in this thesis is unique in several aspects which relate to developing applications to run in real-time on an autonomous marine platform. We describe the implementation of a particle filter that focuses on tracking submerged contacts using applications developed in MOOS-IvP, a powerful open-source framework for marine autonomy research. The work done with MOOS-IvP so far has focused on scientific research including tracking environmental features such as plumes [30] or performing search tasks for stationary objects like mines [33]. There has been no MOOS-IvP implementation for localizing and tracking submerged contacts by way of particle filters.

A second unique aspect of this thesis is the use of adaptive behaviors to maneuver the ASV based on the particle filter's estimates. Significant work has been done on adaptive range tracking of underwater contacts [41] through filters, however much of this work has yet to be adapted to ASVs or AUVs. This thesis leverages previous work done on adaptive range tracking techniques and implements them on an ASV using a particle filter which produces a hypothesis of the contact's course, speed, and three-dimensional position. Based on that information, the ASV maneuvers to either intercept or perform target motion analysis (TMA) to produce a better hypothesis.

Whereas much of the research that has been done with particle filter tracking of submerged contacts deals with the use of bearing-only information [35], our work focussed on the use of range-only information, which presents unique challenges. Bearing-only tracking can achieve system observability by having three independently measured bearings from a non-maneuvering target [17], if ownship's trajectory has at least one more nonzero derivative than the target [5], or ownship's motion consist of a series of course change maneuvers steadying up on constant velocity segments [26]. In the range-only case, the question of maneuvering to achieve observability can be more difficult, and is discussed further in Chapter 4.

Target tracking for autonomous robots has been extensively researched for a variety of application domains. Previous research in autonomous vehicle tracking has focused on Unmanned Aerial Vehicles (UAVs) employing a fusion of multiple sophisticated sensors including LIDAR, RADAR, laser range finders, and visual. However, the tracking process is a significant challenge when it is limited to a single source of information, in this case range. Range-only information is difficult to use for tracking because global observability is not guaranteed [31]. Cooperation of multiple tracking vehicles is an obvious approach to overcoming limited observability, however the problem of cooperative autonomous tracking of submerged adversarial agents has not been extensively researched in the marine robotics research community.

Another significant challenge in this thesis work, in comparison to the related problem of AUV localization [18, 21, 22], is the number of unknown variables. For cooperative localization [16], one typically knows the heading and depth of the target vehicle, which are transmitted to the tracking vehicle via acoustic modems. In contrast, in tracking a non-cooperative target, the course, speed, depth, and initial position are unknown.

1.6 Thesis Contribution

The primary contribution of this thesis is a developed MOOS-IvP application that is capable of receiving range only information and through the combination of TMA techniques and a particle filter, can track a submerged contact within the restrictions of Section 1.2. Our work shows that, under certain conditions, the particle filter combined with TMA techniques was capable of detecting, localizing, and tracking a maneuvering contact through with range-only information.

The criteria for success on the contributions of this thesis are defined as the following:

1. Create a MOOS-IvP software application that would allows a user to specify a set of initial conditions that enables tracking of a submerged contact.
2. Conduct a series of simulation runs and in-water experiments to verify the MOOS-IvP software used by the ASV is able to autonomously track a single simulated submerged contact.

1.7 Thesis Overview

The remainder of this thesis is organized by the following chapters:

Chapter 2 provides an overview of the particle filter and how it was used for this thesis.

Chapter 3 provides an overview of the software developed with MOOS-IvP.

Chapter 4 provides an overview on the difficulty of using target motion analysis for range-only information.

Chapter 5 explores the particle filter's settings and the performance of the application pParticleFilterAP in localizing and tracking of submerged contacts. This chapter also shows the results from both the simulation and a real in water test.

Chapter 6 presents the conclusions of this thesis and discusses potential topics for future work.

Chapter 2

Particle Filter

In this project, a particle filter was used and investigated to determine the feasibility of localizing and tracking a contact using a singular vehicle with a single range only sensor. A particle filter was chosen for its ability to track contacts that exhibit non-Gaussian and nonlinear behaviors that are typical of sea going vessels. Sea going vessels also have the ability to maneuver, which can be detected by a particle filter by observing the shift in the weighted particles. A particle filter is able to achieve this by approximating the posterior with a finite number of parameters [36], in this case range. It does this by updating the probability distribution recursively through a two-step process. The first step happens when the state change of the particle is applied, and the second step, happens when the new sensor data are received and used to update the weight of the particles [4].

2.1 Particle Filter Overview

Figure 2-1 shows the basic structure of the particle filter algorithm used in this project.

The setup for the particle filter is as follows [1, 15, 36]. The first assumption is that there are a series of range only sensors that are capable of detecting the contact. The contact's state and particle's state are given by:

$$\zeta_t = [x \; y \; z \; \theta \; \Theta \; v] \tag{2.1}$$

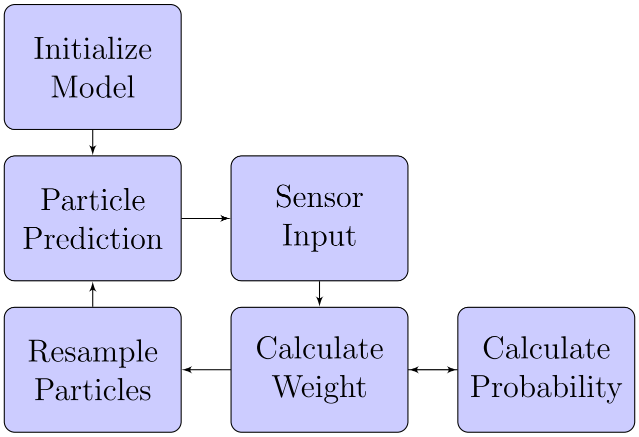


Figure 2-1: Particle Filter Flow Chart: Particles are initialized on the onset of the first range measurement. Particles are continually updated and compared to sensor input; based on this difference a particle's weight is determine based on its probability. Once the combined weight sum drops below a certain threshold, the set of particles are resampled with replacement to form a new set of particles.

where x,y, and z are the Cartesian coordinates of the contact, θ is the contact's course, Θ is the contact's pitch angle, and v represents the contact's speed. In this case the random variables are θ, Θ, and v. The observation is a vector of ranges given by:

$$r_i^t \quad = \quad [r_1 \; r_2 \; ... \; r_n] \tag{2.2}$$

where r_i^t is the sensed range from each of the i sensors at time t. The particle filter works by taking a set of data, range in this case, and creating a sample of random *particles* that satisfies the data,

$$\zeta_t^i \sim q(\zeta_t^i|\zeta_{t-1}^i, r_t) \tag{2.3}$$

where ζ_t^i represents an individual particle i at time t. In the case of tracking a contact, the range is used to produce particles that have a (x,y,z) coordinate from a random bearing angle and elevation angle. A random course, pitch, and speed are also generated to determine the translation motion of a particle traveling from time t to t+*1*. The equations below show how the x, y, and z coordinates are determined from the random bearing ϕ and random elevation angle Θ.

$$x = r \cos\phi \sin\Theta \tag{2.4}$$

$$y = r\ sin\phi \sin\Theta \tag{2.5}$$

$$z = r \cos\Theta \tag{2.6}$$

$$r = \sqrt{x^2\ +\ y^2\ +\ z^2} \tag{2.7}$$

2.2 Initialization

Figure 2-2 shows a field of particles of radius R, where the blue points represent the particles, the yellow point represents ownship, and the red point is an example particle. Since the only information that is given is range, the particles are initially randomly dispersed at a distance R from ownship, with a bearing from ownship of ϕ, and an elevation angle of Θ (Algorithm 6 lines 3-4). Initially, both a course of θ_i and a speed of v_i are uniformly generated (Algorithm 6 lines 3-4). It is important to note that the contact could (but not necessarily) be one of the blue dots. At this point, t=0, each particle has an equal weight for representing the belief state.

Algorithm 1 Particle Filter: Initialization

```
1:  for $i = 1 : N$ do
2:      Sample $\zeta_t^i \sim q(\zeta_t^i | \zeta_{t-1}^i, r_t)$
3:      Generate Random $\phi^i$
4:      Generate Random $\Theta^i$
5:      Generate Random Course $\theta^i$
6:      Generate Random Speed $v^i$
7:      $x^i = r_t \cos\phi^i \sin\Theta^i$
8:      $y^i = r_t \sin\phi^i \sin\Theta^i$
9:      $z^i = r_t \cos\Theta^i$
10: end for
```

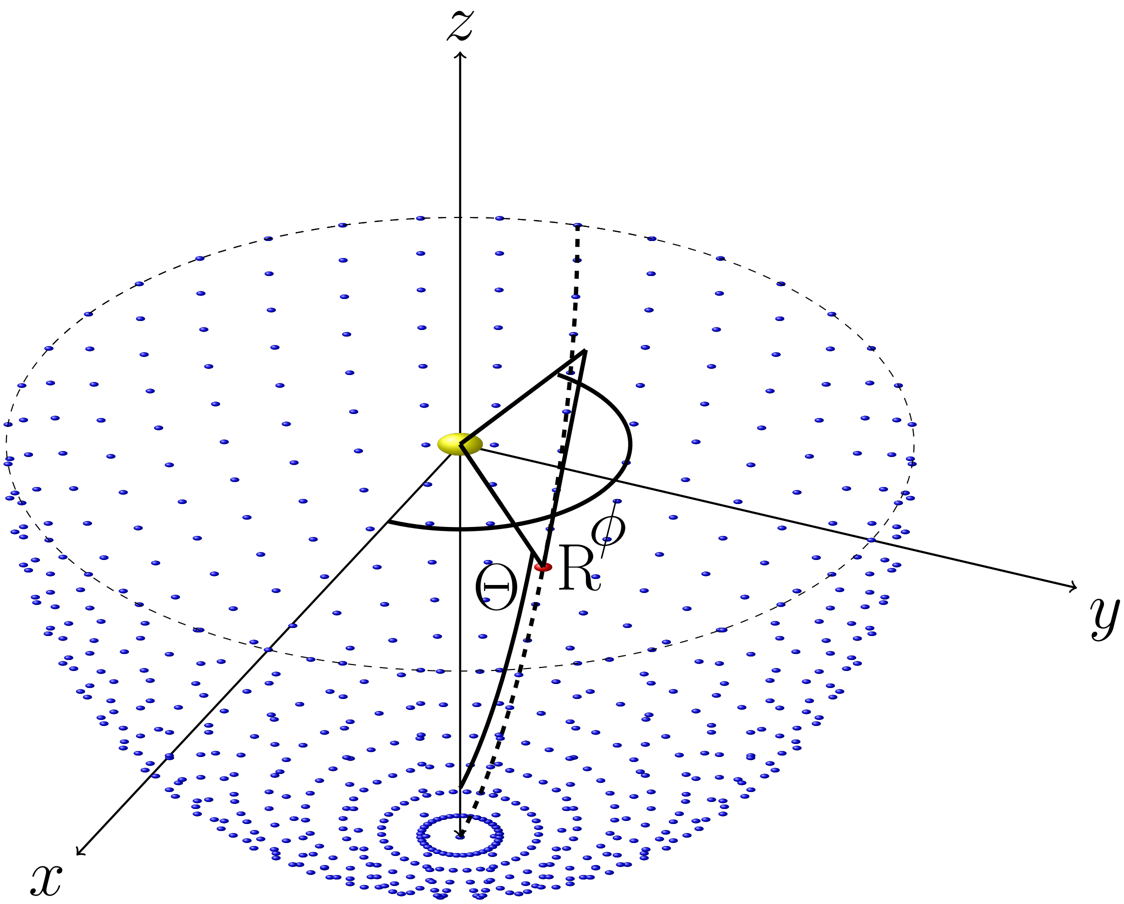


Figure 2-2: Particles are initially generated from an initial range measurement. The yellow circle represents ownship and at a distance R away are particles represented by blue circles. Each particle is given a random course, speed, and position based on the initial range. In this case the range is less than the bottom depth. In the case where bottom depth was less than range, the particle filter is truncated to compensate for bottom depth.

2.3 Particle Prediction

In each cycle, the state of the particle is predicted based on the previous state, course, and speed. One important distinction in the prediction step is the addition of noise for both course and speed variables. To propagate particle diversity a noise term was added to the random variables, in this case v_{noise} and θ_{noise}. One way to think about this process is that the resampling step disperses the particles in the vicinity of the predicted contact of interest with the goal of obtaining a better solution. The noise is used to create this dispersion of particles.

$$v = v + v_{noise} \tag{2.8}$$

$$\theta = \theta + \theta_{noise} \tag{2.9}$$

$$x^i_{t+1} = x^i_t + v\Delta t \cos\theta \sin\Theta \tag{2.10}$$

$$y^i_{t+1} = y^i_t + v\Delta t \sin\theta \sin\Theta \tag{2.11}$$

$$z^i_{t+1} = z^i_t + v\Delta t \cos\Theta \tag{2.12}$$

Algorithm 2 Particle Filter: Prediction

```
for $i = 1 : N$ do
    $v = v + v_{noise}$
    $\theta = \theta + \theta_{noise}$
    $x^i = x^i + v\Delta t \cos\theta \sin\Theta$
    $y^i = y^i + v\Delta t \sin\theta \sin\Theta$
    $z^i = z^i \cos\Theta$
end for
```

2.4 Sensor Input

Particle filters are capable of receiving various types and numbers of inputs. This thesis explores using a particle filter with only one range sensor input. In section 4.2.3 multiple range inputs were used from multiple vehicles to demonstrate the convergence of the particles.

2.5 Determination of Weights

Once the sensor information was received, the next step was to determine the weight of the particles (importance factor), w. For each new range measurement the weight, w_t^i, was found using the following equation: [36]

$$w_t^i = w_{t-1}^i \frac{p(r_t|\zeta_t^i)p(\zeta_t^i|\zeta_{t-1}^i)}{q(\zeta_t^i|\zeta_{0:t-1}^i, r_t)} \tag{2.13}$$

When the transition prior is used as the importance function, $p(\zeta_t^i|\zeta_{t-1}^i) = q(\zeta_t^i|\zeta_{0:t-1}^i, r_t)$, the weight simplifies to $w_t^i = w_{t-1}^i p(r_t|\zeta_t^i)$. The weights are then normalized so that $\sum_i w_i^t = 1$.

The probability function, $p(r_t|\zeta_t^i)$, was solved by understanding and identifying the sources of error for the range measurement. Three sources of error identified: the first was range measurement error produced by the range sensor, or in this case range simulator. The second was the GPS error since the position of the contact was determined by both the sensed range and the position of ownship. The third error was the latency error from the vehicle's internal communications and processes. Some time delay existed from receiving the sensor information to the processing of the information to sharing the information with the various MOOS modules. These errors and delays were significant enough to cause error in the particle filters prediction, resulting in the predicted position being several meters from the contact's position. By using the Central Limit Theorem [25], if S_n is the sum of n mutually independent random variables; in this case range error, location error, and latency error, than S_n can be approximated as a Gaussian distribution, which is given by the following equation:

$$p(r_t|\zeta_t^i) = f_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-\mu)^2/(2\sigma^2)} \tag{2.14}$$

Figures 2-3 and 2-4 provide illustrations of this distribution. A floor was used in the Gaussian distribution, to help prevent highly weighted particles from being zeroed out by outliers.

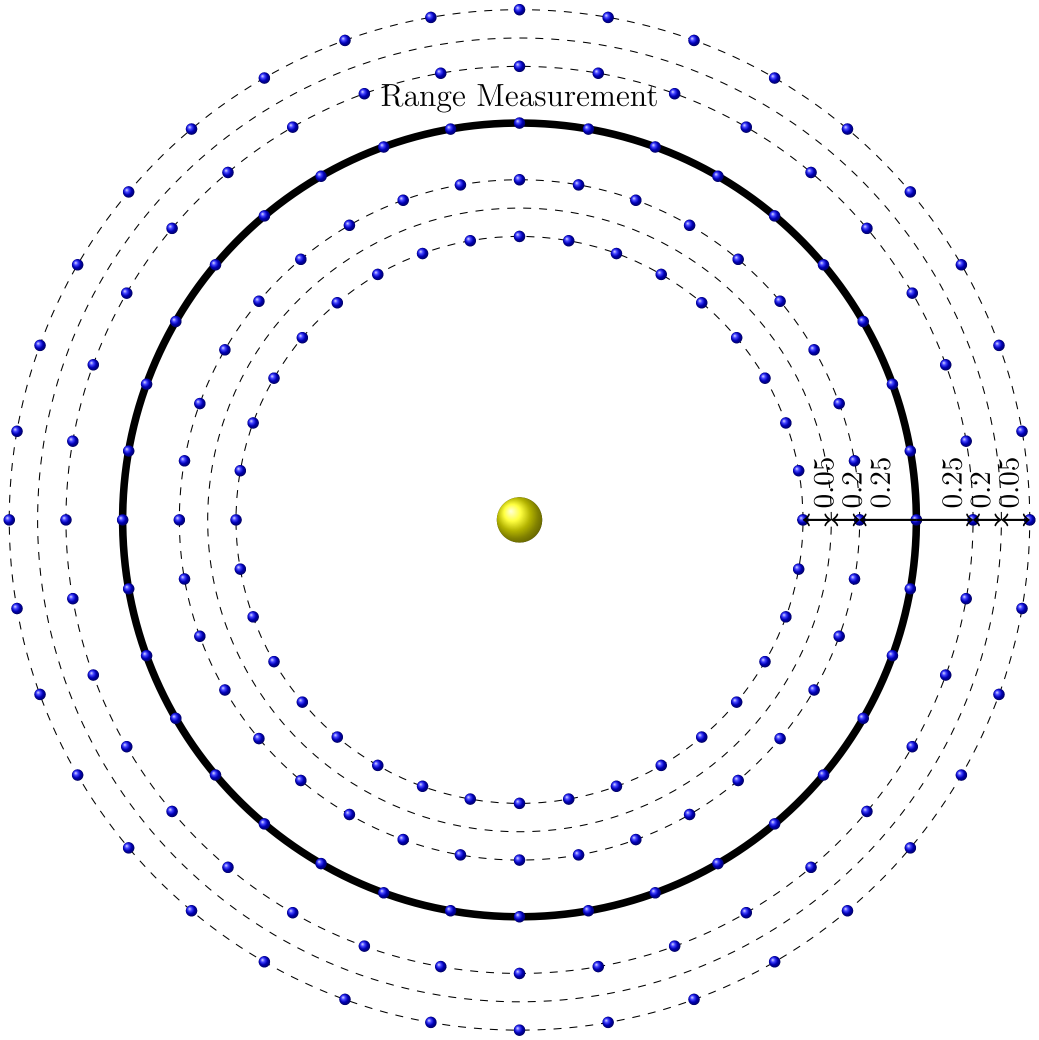

Range Measurement
0.05
0.2
0.25
0.25
0.2
0.05

Figure 2-3: An Example Probability Distribution for a range difference between the actual sensed range (the thick black line) and the particles (the blue circles).

Algorithm 3 Particle Filter: Weight Determination

```
for $i = 1 : N$ do
    Calculate Expected Range of Particle
    Find the Range Difference Between the Expected and Actual, $\Delta Range$
    Find $P(\Delta Range)$
    $Weight^i = Weight^i * P(\Delta Range)$
    Overall Weight = Overall Weight+$Weight^i$
end for
NORMALIZE WEIGHT
for $i = 1 : N$ do
    $Weight^i = \frac{Weight^i}{OverallWeight}$
end for
```

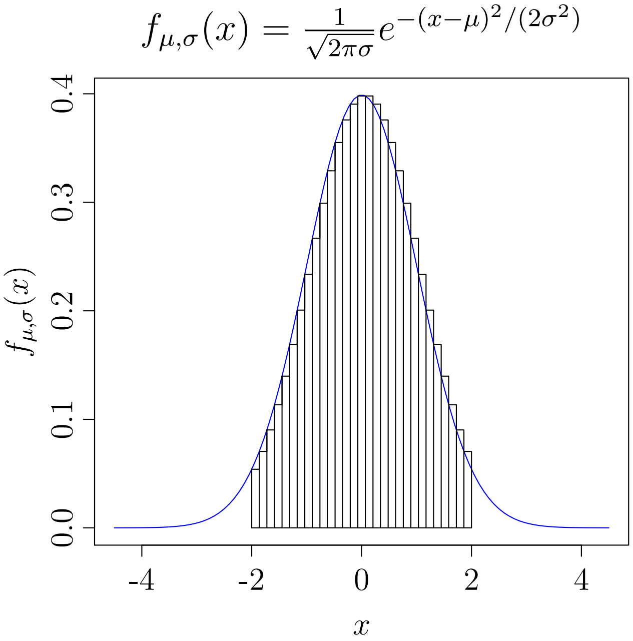


Figure 2-4: A Gaussian Distribution was used to estimate the probability of the range difference between a particle and the actual range measurement.

2.6 Resampling

After the weight of the particles were calculated, the particle filter enters the resampling process [1, 15, 36]. This step is significant because it helps avoid degeneration of filter's estimate ζ_t while continuing to localize and track the contact. To do this the particle filter resamples the entire set of particles once $M_{eff} < M_{threshold}$, where $M_{eff} = \frac{1}{\sum_{L=1}^{i}(w_k^{(L)})^2}$ [19] and $M_{threshold} = \frac{N}{2}$. The particle algorithm draws with replacement from the N particles in the set ζ_t. The probability of drawing each particle is determined by the weight that was previously calculated. This resampling transforms the set of N particles to another set of N particles but with a new distribution based on the previous weights. This resample shifts the initial probability density function (pdf) to a pdf that is more representative of the problem, Figure 2-5 is an example of the shift in pdf.

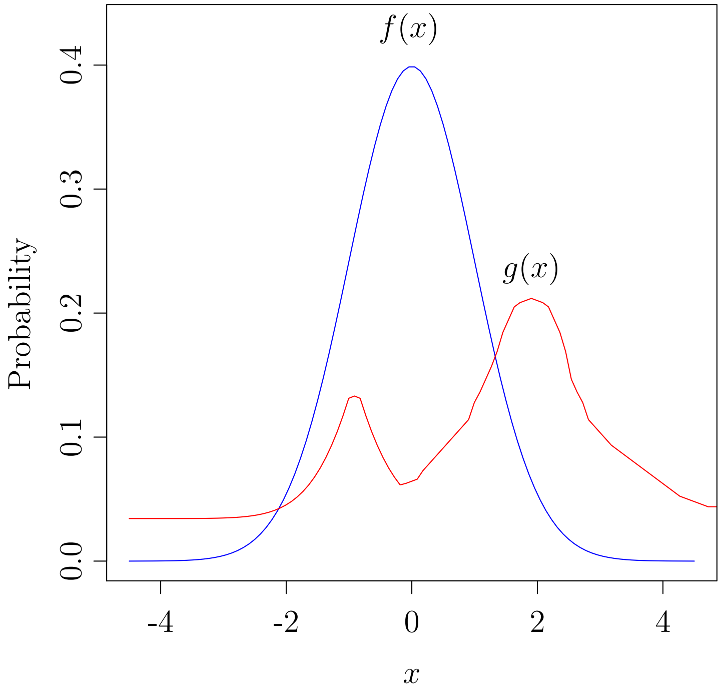


Figure 2-5: An initial PDF was assumed (f(x)), once actual measurements are taken the PDF shifts as a result of resampling to a PDF that reflects the actually probability of the system (g(x)).

2.7 Particle Filter Momentum

One of the difficulties in tracking a maneuvering contact is how the particle filter overcomes the momentum of the particles. For example, if a contact suddenly changed

Algorithm 4 Particle Filter: Resampling

```
if $M_{eff} < M_{threshold}$ then
    for $i = 1 : N$ do
        Draw With Replacement from $\zeta_t^i$
    end for
end if
```

its course by 120° the particle filter would have a difficult time adapting to that change. Recall that after each update, velocity and course are updated as follows $v = v + v_{noise}$ and $\theta = \theta + \theta_{noise}$. If the course noise was set to allow a deviation of 30°, it would take a minimum of four updates to have a particle at the new course at which the contact would have a significant head start from the group of particles. Once the particles have reached the correct course, the particles would have to have a higher speed than the contact in order to catch up to the contact. Once the particles caught up to the contact they would overshoot, hence, the momentum problem for particle filters results. Figures 2-6 and 2-7 are examples of a sharp maneuver where the particles do not follow.

One of the ways to overcome the momentum problem is by increasing the course and speed noise during the update step. The disadvantage with this method is that when the noise is increased so is the associated range error.

Another method to mitigate this problem and the one incorporated in this thesis is the concept of reserve particles. The purpose behind reserve particles is to take a small portion of the overall set and use them to prevent the momentum problem from happening. Reserve particles do this in the resample portion of the algorithm. When resampling occurs the particles are drawn from the same sample set of X, Y, and Z coordinates but their course and speed are generated randomly instead of drawn from the sample. By using a randomly generated course and speed at this point, losing track and overshooting particles are reduced. One disadvantage with reserve particles is that it creates an offset in the average number of particles as a result of the reserve particles be dispersed uniformly from the resample point. For this reason it is important to limit the number of reserve particles to less than 10 percent of the overall number of particles.

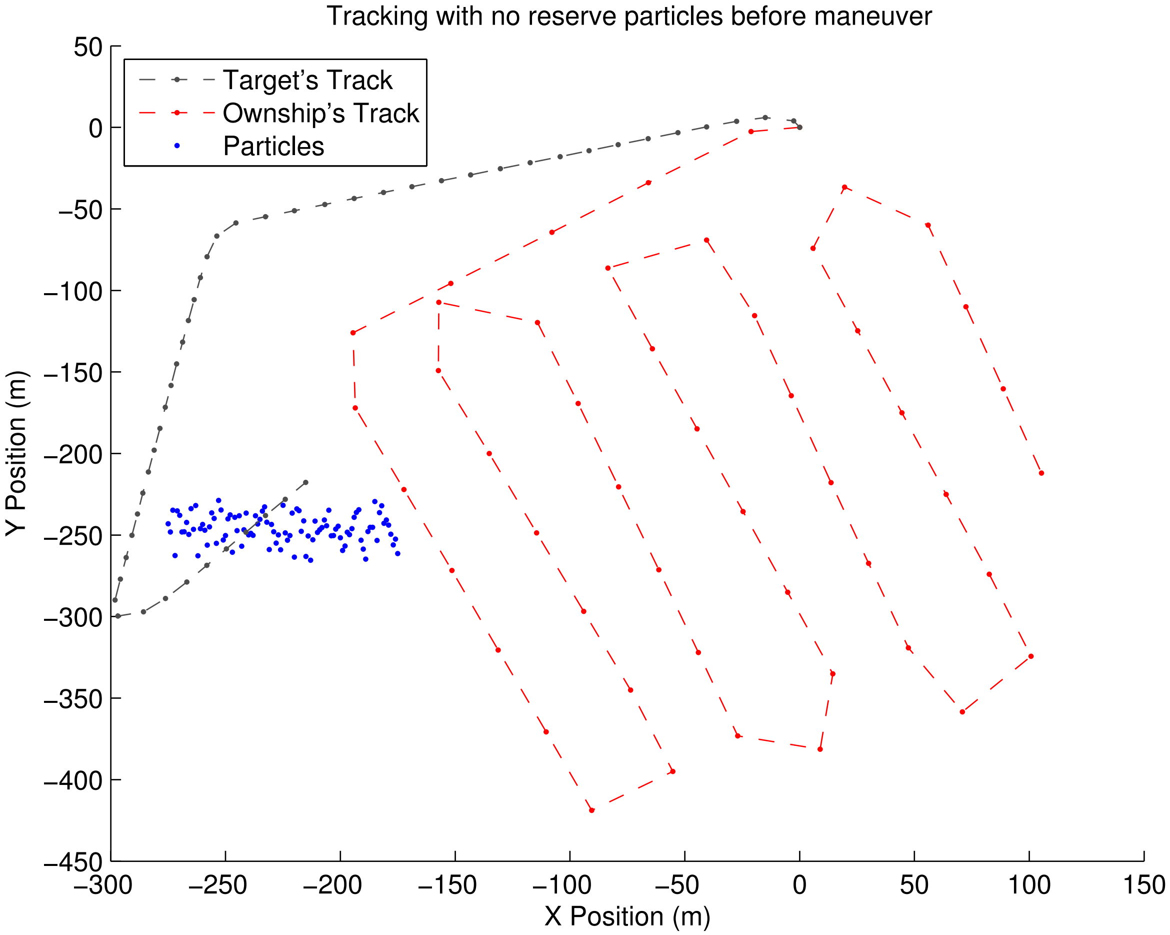


Figure 2-6: The particles have consolidated and grouped around the target's track. If the contact continues with its current course and speed the particles will continue to track.

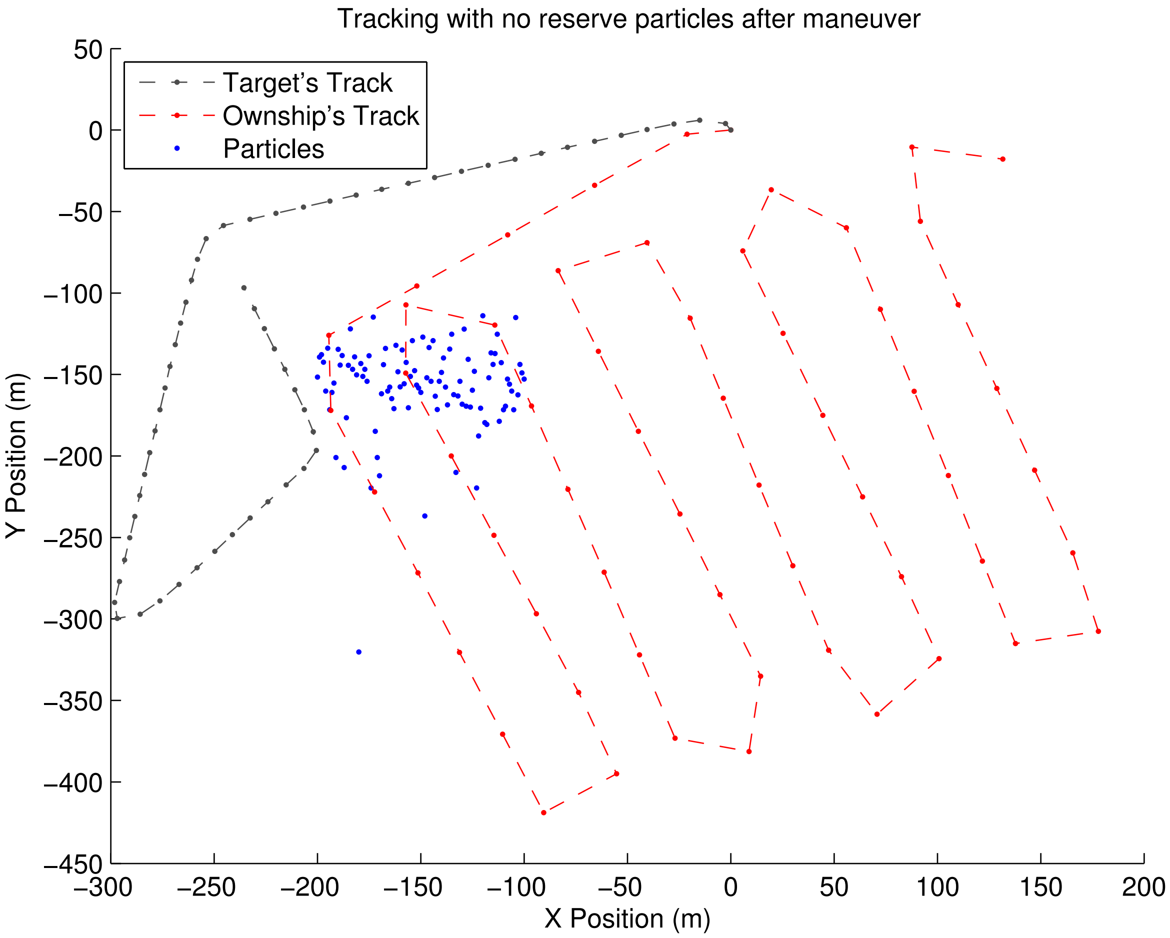


Figure 2-7: Momentum of Particles: As the target maneuvers, the particles are slow to regain the target's track. The particles initial continue on the previously projected track, the outliers close to the contact slowly gain weight and eventual resampling occurs to shift the particles closer to the actually track. If the particles deviated too far from the track they may not reconverge on the contact and instead converge on track with a parallel solution.

Chapter 3

Software Architecture and Modules

The principle work developed by this project was done using MOOS-IvP. A particle filter software module was created (pParticleFilterAP) to hypothesize possible contacts' course, range, and position information. These hypotheses were sent to the IvP Helm CutRange Behavior which maneuvered the ASV autonomously in order to track the contact of interest.

3.1 MOOS

One advantage of using MOOS-IvP as a software suite is that it is platform-independent, meaning that it can be run on any number of ASVs, AUVs, or other autonomous vehicles. MOOS is structured to act as a communicator for different processes: each process represents a different application, component, or attribute of the Autonomous Vehicle (AV). MOOS-IvP has several central applications that are open-source and platform-independent but also allows, the user has the option of extending the software by writing new applications. Inter-process communication in a community of MOOS applications flow through a central MOOS Database (MOOSDB). Each process is independent of one another and they share their information with MOOSDB. The processes share information by publishing a (Message Key, Message Value) pair to the MOOSDB. Other processes are able to request information from the MOOSDB structure by subscribing for needed information. Figures 3-1 through 3-4 illustrate how several MOOSDB communities exist in order to perform different tasks and communicate with one another. Figure 3-1 shows that the individual AVs communicate directly to the shoreside. The shoreside MOOS community receives these messages via the pMOOSBridge application and posts the messages to the local shoreside MOOSDB. This communication is done through WiFi connectivity. One important aspect is that each MOOSDB community is located locally, meaning that the AVs' communities are located on the computers within the AVs and that the shoreside community is located on a computer shoreside. For the test, the submerged contact was simulated on a kingfisher.

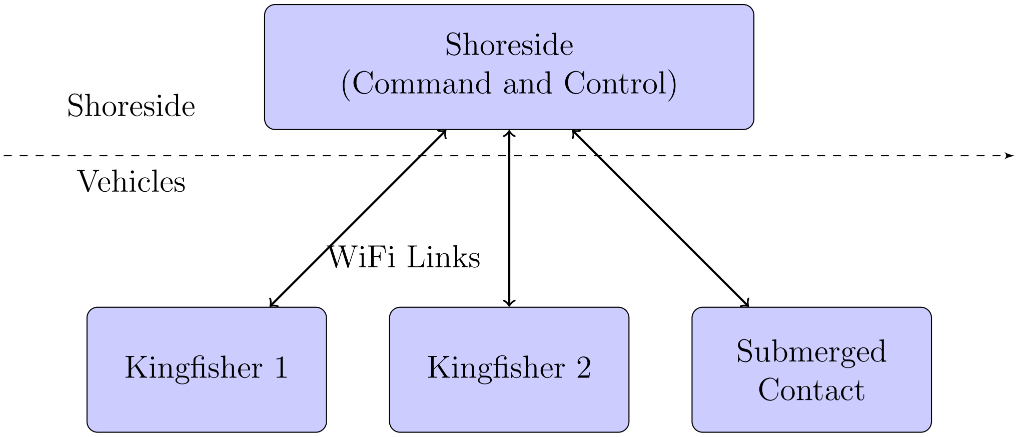


Figure 3-1: MOOS Shoreside to Multi-Vehicle Topology: Three vehicles are deployed with each vehicle maintaining a baseline connectivity with the shoreside command and control through WiFi communications. Each vehicle node and shoreside node are comprised of a dedicated MOOS community. In our case only two vehicles were present, leaving the need to simulate one. [9]

Figures 3-2 through 3-4 were the simulated MOOS communities used for this project.

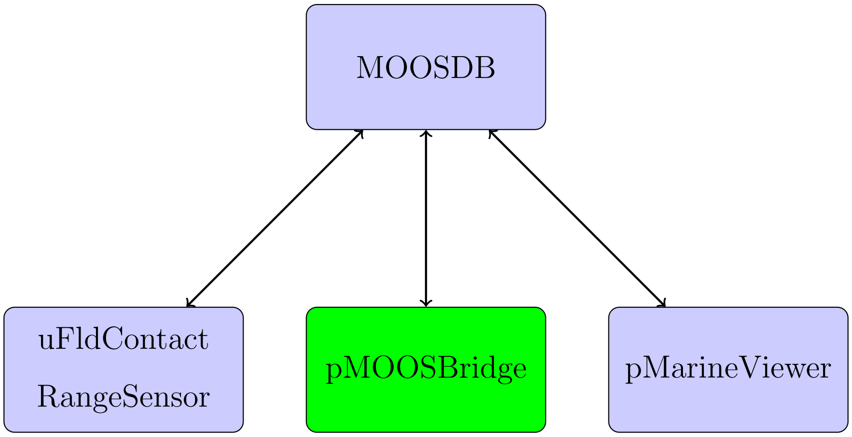

MOOSDB
uFldContact RangeSensor
pMOOSBridge
pMarineViewer

Figure 3-2: MOOS Shoreside Community: This is an example of the simulated community used for the shoreside.

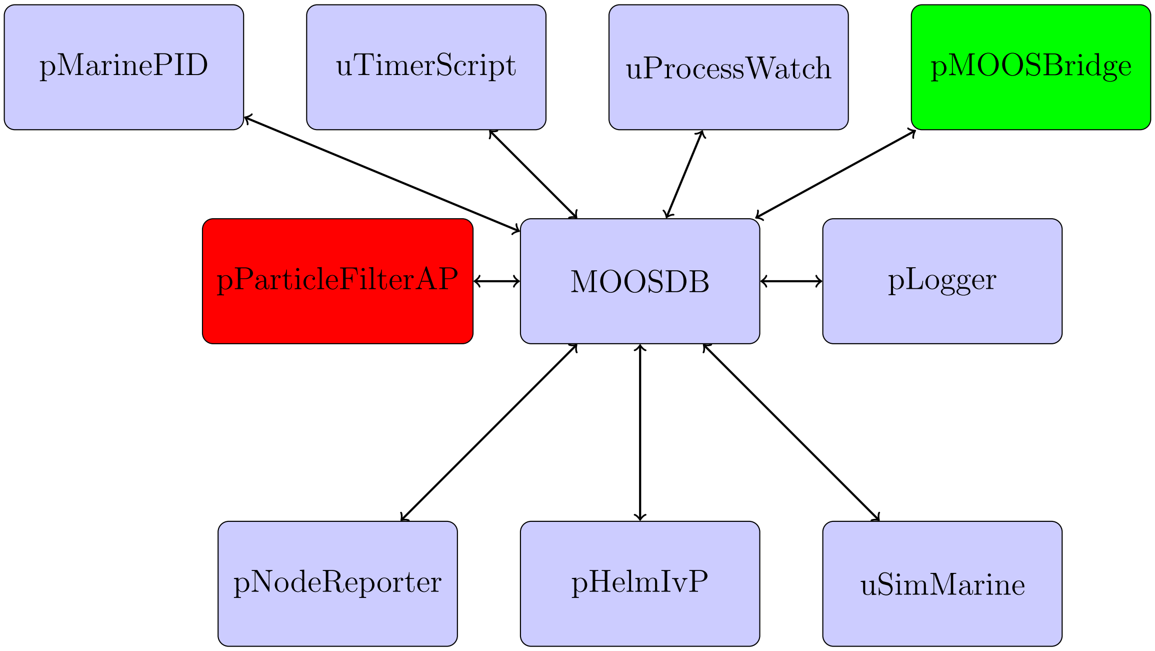

pMarinePID
uTimerScript
uProcessWatch
pMOOSBridge
pParticleFilterAP
MOOSDB
pLogger
pNodeReporter
pHelmIvP
uSimMarine

Figure 3-3: MOOS ASV Community: This is an example of the simulated community used for the ASV. pParticleFilterAP was the application used for localizing and tracking the contact.

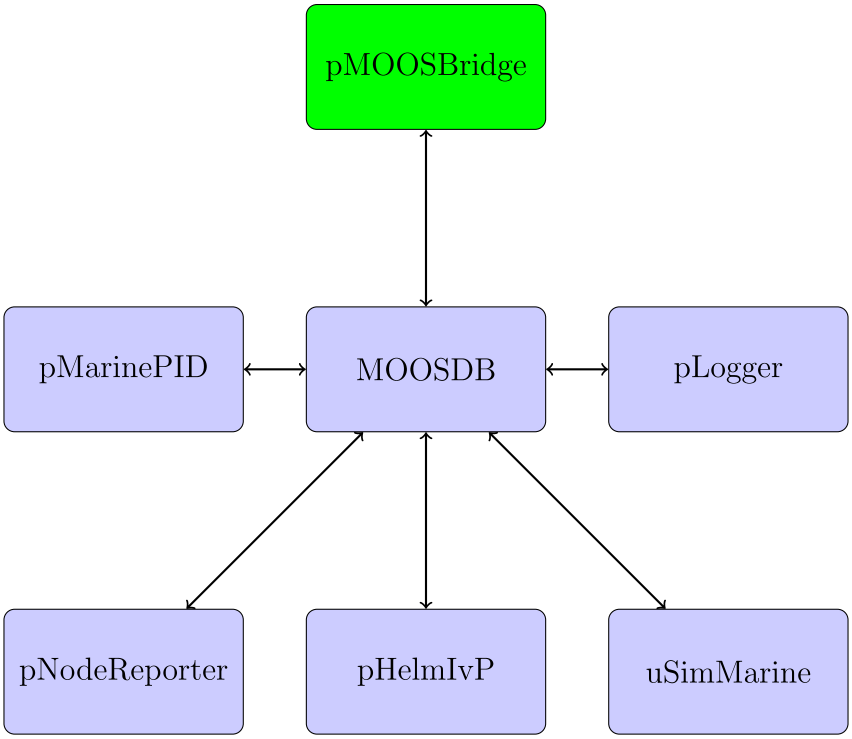


Figure 3-4: MOOS Submerged Contact Community: This is an example of the simulated community used for the submerged contact.

The following are descriptions of the MOOS Modules [11] shown in Figures 3-2 through 3-4;

- MOOSDB is the central database used for MOOS modules.
- pLogger is a process that records the publications of applications involved in a MOOS session.
- pHelmIvP is a behavior-based autonomous decision-making MOOS application. It consists of a set of behaviors reasoning over a common decision space such as the vehicle heading and speed. Behaviors are reconciled using multi-objective optimization with the Interval Programming (IvP) model. It publishes information such as desired heading, speed, and depth to drive the vehicle.
- uSimMarine is a simple 3D vehicle simulator that updates vehicle state, position and trajectory, based on the present actuator values and prior vehicle state.

Typical usage scenario has a single instance of uSimMarine associated with each simulated vehicle.

- pNodeReporter is a simple MOOS app for collecting node information such as present vehicle position, trajectory and type, and posting it in a single report for sharing between vehicles or sending to a shoreside display.
- pEchoVar is tool that may be used to subscribe for a variable and re-publish it under a different name. It also may be used to pull out certain fields in string publications consisting of comma-separated parameter=value pairs, publishing the new string using different parameters.
- pMarinePID is an application that provides proportional-integral-derivative control for vehicles parameters such as depth-pitch, speed-thrust, and heading-rudder.
- uTimerScript allows the user to script a set of pre-configured pokes to a MOOSDB with each entry in the script happening after a specified amount of time. The script may be paused or fast-forwarded. Events may also be configured with random values and happen randomly in a chosen window of time.
- uFldContactRangeSensor is typically run in a shoreside MOOS community. It takes reports from remote vehicles and notes their position. It takes a range request from a remote vehicle and returns a range report indicating that vehicle's range to nearby vehicles. Range requests may or may not be answered dependent on inter-vehicle range. Reports may also have noise added to their range values.
- uProcessWatch monitors the presence of MOOS processes by subscribing to and identifying changes in clients that are connected to MOOSDB.
- pMarineViewer is a GUI tool for rendering events in an area of vehicle operation. It repeatedly updates vehicle positions from incoming node reports, and will render several geometric types published from other MOOS apps. The viewer

may also post messages to the MOOSDB based on user-configured keyboard or mouse events.

- pMOOSBridge is used to communicate information between distinct MOOS communities. These communities may be all on the same machine or distributed on network.
- pParticleFilterAP generates particles that are used to localize and track contacts.

3.2 IvP Helm

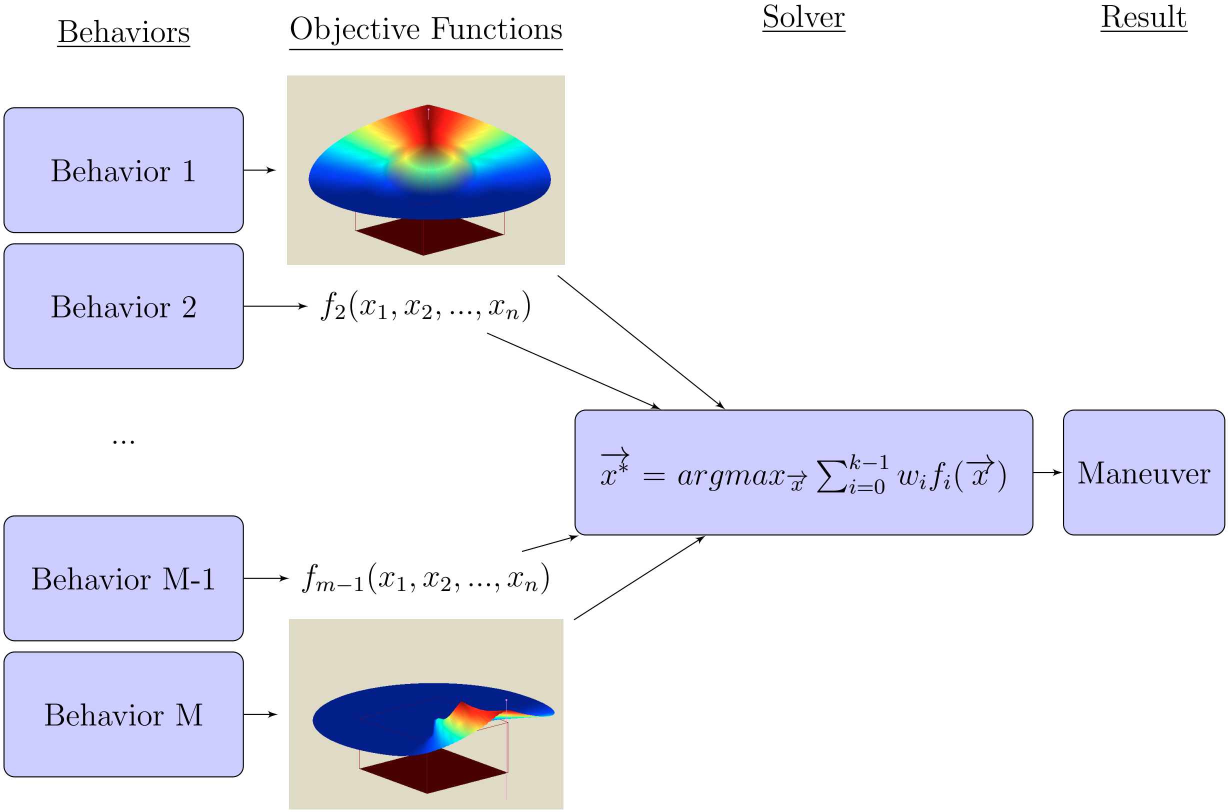

Behaviors
Objective Functions
Solver
Result
Behavior 1
Behavior 2
$f_2(x_1, x_2, ..., x_n)$
...
$\overrightarrow{x^*} = argmax_{\overrightarrow{x}} \sum_{i=0}^{k-1} w_i f_i(\overrightarrow{x})$
Maneuver
Behavior M-1
$f_{m-1}(x_1, x_2, ..., x_n)$
Behavior M

Figure 3-5: Action Selection with Interval Programming: Each active behavior produces an objective function. The objective functions feed into a solver which handles multiple object functions from each behavior by multi-objective optimizations using the interval programming method. [8]

The IvP Helm [11] is a behavior-based autonomous helm implemented as a MOOS application.

It is based on Interval Programming [8] where each active behavior produces a piecewise linearly defined objective function. On each iteration of the Helm, the IvP solver (Figure 3-5) solves a multi-objective optimization problem from the collection of weighted objective functions from contributing behaviors. The result on each iteration, is a requested heading, speed and depth to be published to the MOOSDB. Figure 3-5 demonstrates the flow path from the initial behavior to the maneuver. The red areas represent the optimal courses and speeds based on the objective functions.

3.3 Range Sensor Information

A simulated off-board range measurement is produced through the application uFldContactRangeSensor [9]. The parameters for this module are shown in Configuration Block 5. The range measurements produced by this module are based on the ground truth value with additive noise. For our work in this project, additive noise with either a uniform or a Gaussian error distribution was utilized.

This application produces a range report with the following format that is read by the particle filter:

CRS_RANGE_REPORT =”name=archie,range=23.4,target=jackal,time=2342551.213”

- Line 8: ping_wait is the time delay between range pulses.
- Line 13: rn_algorithm is the amount of error introduced to ground truth. In this case the error that was introduced was from a uniform distribution, which has a maximum error of 4 percent of the actual range measurement.

3.4 pParticleFilterAP

This section provides guidance for how to operate the particle filter settings developed for this thesis. The algorithms and descriptions used within the particle filter are

Configuration Block 5 uFldContactRangeSensor Settings

```
ProcessConfig = uFldContactRangeSensor
{
AppTick = 4
CommsTick = 4

reply_distance = jackal = 50
reach_distance = archie = 390
ping_wait = archie = 5
report_vars = both
ping_color = white
reply_color = chartreuse
verbose = true
rn_algorithm = uniform,pct=0.04
}
```

found in Chapter 2. Configuration Block 6 is one possible way to configure the pParticleFilterAP.

3.4.1 Configuration Parameters for pParticleFilterAP

- Line 6: XYZ_REPORT is used to obtain local AV information such as course, speed, position, and time. When a collaborative contact is used, this report is also used to get the collaborator's time and position to use with the range report.
- Line 7: RANGE_REPORT is used to extract both the ownship's range report and collaborative ship's range report.
- Line 8: MY_SHIP used to identify the ownship.
- Line 9: MY_FRIEND used to identify the collaborating ship.
- Line 10: MY_CONTACT used to identify the contact that is being tracked.
- Line 11: MY_BEST_GUESS and Line 12: MY_AVG_GUESS are solutions generated for the use of the CutRange_BHV. A discussion of these two parameters is located in Section 5.1.1

Configuration Block 6 Particle Filter Settings

```
ProcessConfig = pParticleFilterAP
{
AppTick = 4
CommsTick = 4

XYZ_REPORT = NODE_REPORT
RANGE_REPORT = CRS_RANGE_REPORT
MY_SHIP = archie
MY_FRIEND = betty
MY_CONTACT = jackal
MY_BEST_GUESS = besttarget
MY_AVG_GUESS = avgtarget
MAX_SPEED = 2
MAX_DEPTH = 30
SPEED_NOISE = 0.5
COURSE_NOISE = 10
RANGE_VAR = 10
NO_PARTS = 2000
N_THRESHOLD = 1000
DISPLAY_PARTS = YES
RESERVE_PARTS = 300
CEILING = 4
FLOOR = 100
STORE_DATA = NO
}
```

- Line 13: MAX_SPEED is the maximum possible speed of the contact. It is used to limit the particle space (section 5.1.2).

- Line 14: MAX_DEPTH is the maximum possible depth of the contact based on geographic and vehicle constraints. It is used to limit the particle space (section 5.1.2).

- Line 15: SPEED_NOISE is the amount of speed noise introduced in the prediction step (section 2.3). This number is multiplied by a uniform random distribution [0,1].

- Line 16: COURSE_NOISE is the amount of course noise introduced in the prediction step (section 2.3). This number is multiplied by a uniform random distribution [0,1].

- Line 17: RANGE_VAR is used as the variance in the weight evaluation in Section 2.5.

- Line 18: NO_PARTS is the number of particles used. 2000 was found to be an acceptable number. If this number is reduced, the error is increased as a result of decreasing the chances of finding the solution. If this number is increased, so are the required computer resources.

- Line 19: N_THRESHOLD helps determine the point at which the particles are resampled based on overall weight (section 2.6). Typical values are usually half of NO_PARTS.

- Line 20: DISPLAY_PARTS is used to displays particles with pMarineViewer. This does not work well is the number of particle is greater than 200.

- Line 21: RESERVE_PARTS is the number of reserve particles used. This is discussed in section 2.7.

- Line 22: CEILING is a ceiling used for the Gaussian distribution calculation in section 2.5. This is needed to avoid situations in which wrong solutions outweigh correct solutions based on minor range errors.

- Line 23: FLOOR is a floor used for the Gaussian distribution calculation that was discussed in Section 2.5.

- Line 24: STORE_DATA is used to store particle and the best hypothesis information. The stored date is located in data.txt with the following format: (Time, Ownship Position [X,Y,Z], Ownship Course, Ownship Speed, Contact Position [X,Y,Z], Contact Course, Contact Speed, Particle 1 Position [X,Y,Z], Particle1 Course, Particle 1 Speed, ... ,Particle N Position [X,Y,Z], Particle N Course, Particle N Speed, Particle Best Position [X,Y,Z], Particle Best Course, Particle Best Speed, Particle AVG Position [X,Y,Z], Particle AVG Course, Particle AVG Speed).

3.4.2 MOOS Variables Posted by pParticleFilterAP

The primary output of pParticleFilterAP to the MOOSDB are the following node reports needed for the CutRange Behavior.

- NODE_REPORT publishes the highest weighted particle (MY_BEST_GUESS) and the average weight of the particle (MY_BEST_GUESS). This is used in behaviors like BHV_CutRange.

3.4.3 MOOS Variables Subscribed by pParticleFilterAP

Variables subscribed for pParticleFilterAP are summarize below.

- XYZ_REPORT (NODE_REPORT) is the used to get ownship and collaborative ships' attributes such as course, speed, and position information.

- RANGE_REPORT (CRS_RANGE_REPORT) is used to get range reports from ownship and collaborative ships.

3.5 The CutRange Behavior

The CutRange Behavior (Configuration Block 7) acts to reduce the range from ownship to a specified vehicle given by a node report. In the case of this thesis, the node report that is generated is based on either the average particle position or the highest-weighted particle position. Line 13 PATIENCE: Can either be 0 or 100: 0 represents an straight course to the contact's present position and 100 represents a course along the path to CPA.

Configuration Block 7 CutRange Settings

```
 1: ProcessConfig = BHV_CutRange
 2: {
 3: AppTick = 4
 4: CommsTick = 4
 5:
 6: name = trackbest
 7: pwt = 90
 8: condition = RETURN = false
 9: condition = (DEPLOY = true)
10: PWT_OUTER_DIST = 20
11: PWT_INNER_DIST = 2
12: GIVEUP_DIST = 800
13: PATIENCE = 0
14: EXTRAPOLATE = true
15: ON_NO_CONTACT_OK = true
16: TIME_ON_LEG = 60
17: CONTACT = besttarget
18: }
```

3.6 Overall Software Organization

Information flows from the range sensor to the Particle Filter. The Particle Filter communicates with the TMA function (Chapter 4) to determine the best maneuver and sends that information to the CutRange Behavior. This behavior then influences the course chosen in the IvP Helm.

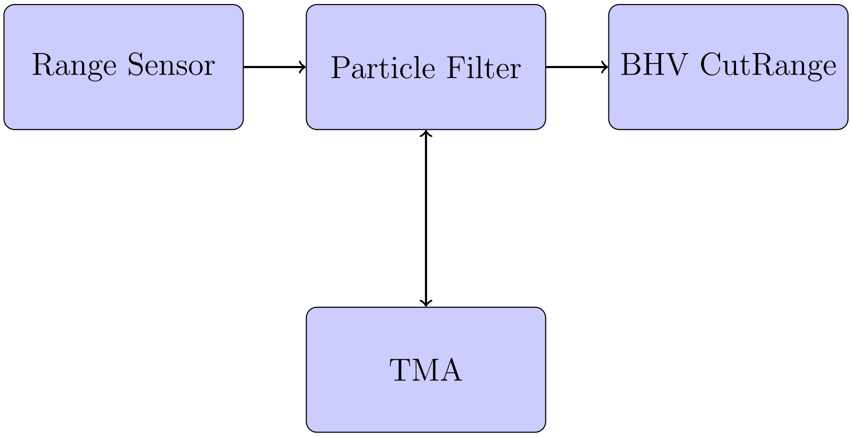

Range Sensor
Particle Filter
BHV CutRange
TMA

Figure 3-6: Software Flow Chart: The main applications used in this thesis are the simulated range sensor, the particle filter with TMA decisions, and the cut range behavior. The particle filter receives input from the simulated range sensors and produce output to the cut range behavior. A similar structure can be found in [41] pg. 303.

Chapter 4

Target Motion Analysis

This chapter discusses the problem caused from the lack of observability encountered when using range-only measurements. A bearing-only measurement solution can be derived through a series of maneuvers [5,14,17,26]; however, there are more limitations for finding system observability of range-only measurement target tracking [34]. This lack of global observability places restrictions and guidelines on how to localize a contact. These guidelines are discussed throughout this chapter.

4.1 Observability Analysis

For simplicity it was assumed that the depth of the submerged contact was constant, thereby reducing the state Equation 4.2 from nine variables to six variables. Figure 4-1 references the relative motion used for these equations. To determine if the system of range-only measurements is observable, the following equations [34] are used:

$$\dot{n} = An + BA_m \tag{4.1}$$

$$n = (X, Y, \dot{X}, \dot{Y}, A_{T_x}, A_{T_y})^T \tag{4.2}$$

$$A_m = (A_{mX}, A_{mY})^T \tag{4.3}$$

$$A = \begin{bmatrix} 0 & I_2 & 0 \\ 0 & 0 & I_2 \\ 0 & 0 & 0 \end{bmatrix} \tag{4.4}$$

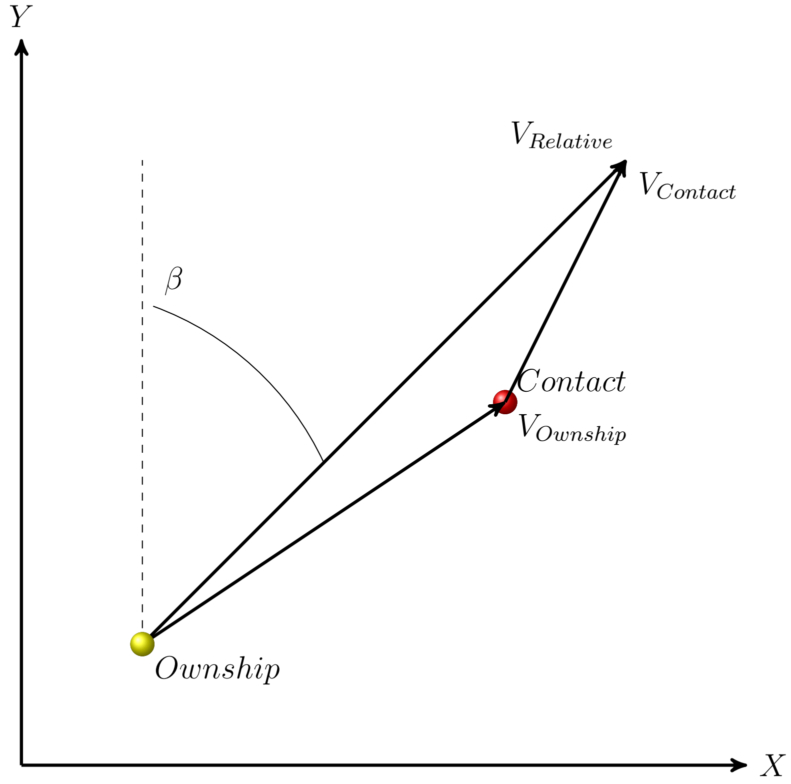
$$B = \begin{bmatrix} 0 \\ -I_2 \\ 0 \end{bmatrix} \tag{4.5}$$

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.6}$$

$$\Delta R = \sqrt{X(t_i)^2 + Y(t_i)^2} \tag{4.7}$$

$$\beta \ = \ tan^{-1}(\frac{Y}{X}) \tag{4.8}$$

Since the value of β is unknown, the contact's position can not be directly solved; this causes the ambiguity that is shown through time progression in Figure 4-2 through Figure 4-5. This ambiguity can lead to non-convergence of the contact. Toward the end of this progression more particles are eliminated; however, several still remain due to the infinite number of course/speed solutions demonstrated in Equations 4.7 and 4.8 above.



Y
$V_{Relative}$
$V_{Contact}$
β
$Contact$
$V_{Ownship}$
$Ownship$
X

Figure 4-1: Relative Motion: Ownship travels at a speed of $V_{Ownship}$ while the contact travels at a speed of $V_{Contact}$. The two important variables needed to be solved for the observability analysis is β and $V_{Relative}$.

– Range Rings in 100 meter increments
▬ Contact's Range
▬ Contact's Track
● Ownship (USV)
● New Particle
● Low Weight Particle
● High Weight Particle

Particle	Course (°)	Speed $\frac{meters}{second}$	Weight
1	225°	0.707	0.10
2	340°	1.068	0.10
3	210.5°	1.300	0.10
4	0°	1.000	0.10
5	137.5°	1.356	0.10
6	235.5°	0.610	0.10
7	180°	0.104	0.10
8	191.9°	1.272	0.10
9	9.9°	1.436	0.10
10	135°	0.342	0.10
Target	337°	1.256	N/A

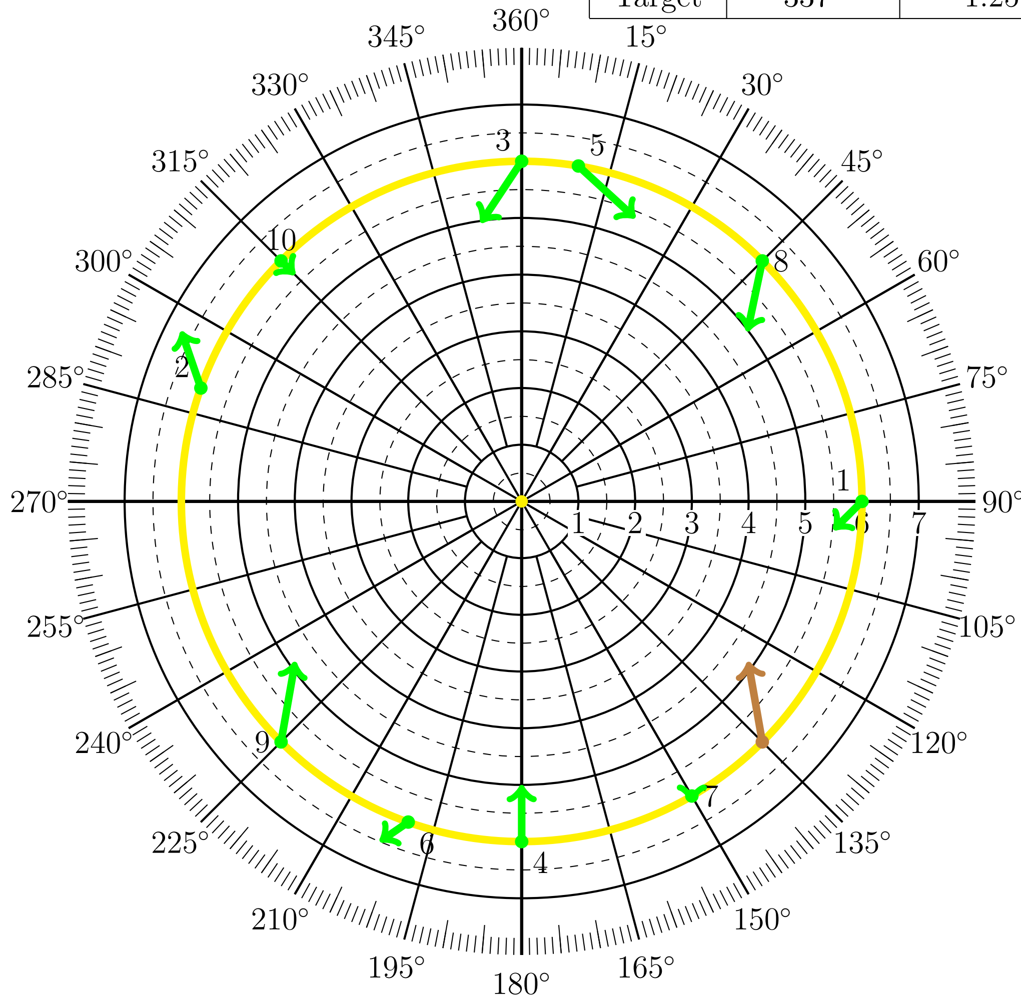


Figure 4-2: Relative Motion Plot of Particles at t = 0. This plot shows the initialization of the 10 particles, each with different courses, speeds, and initial bearings from ownship.

Particle	Course (°)	Speed $\frac{meters}{second}$	Weight
1	225°	0.707	0.068
2	340°	1.068	0.002
3	210.5°	1.300	0.182
4	0°	1.000	0.182
5	137.5°	1.356	0.068
6	235.5°	0.610	0.02
7	180°	0.104	0.068
8	191.9°	1.272	0.182
9	9.9°	1.436	0.182
10	135°	0.342	0.068
Target	337°	1.256	N/A

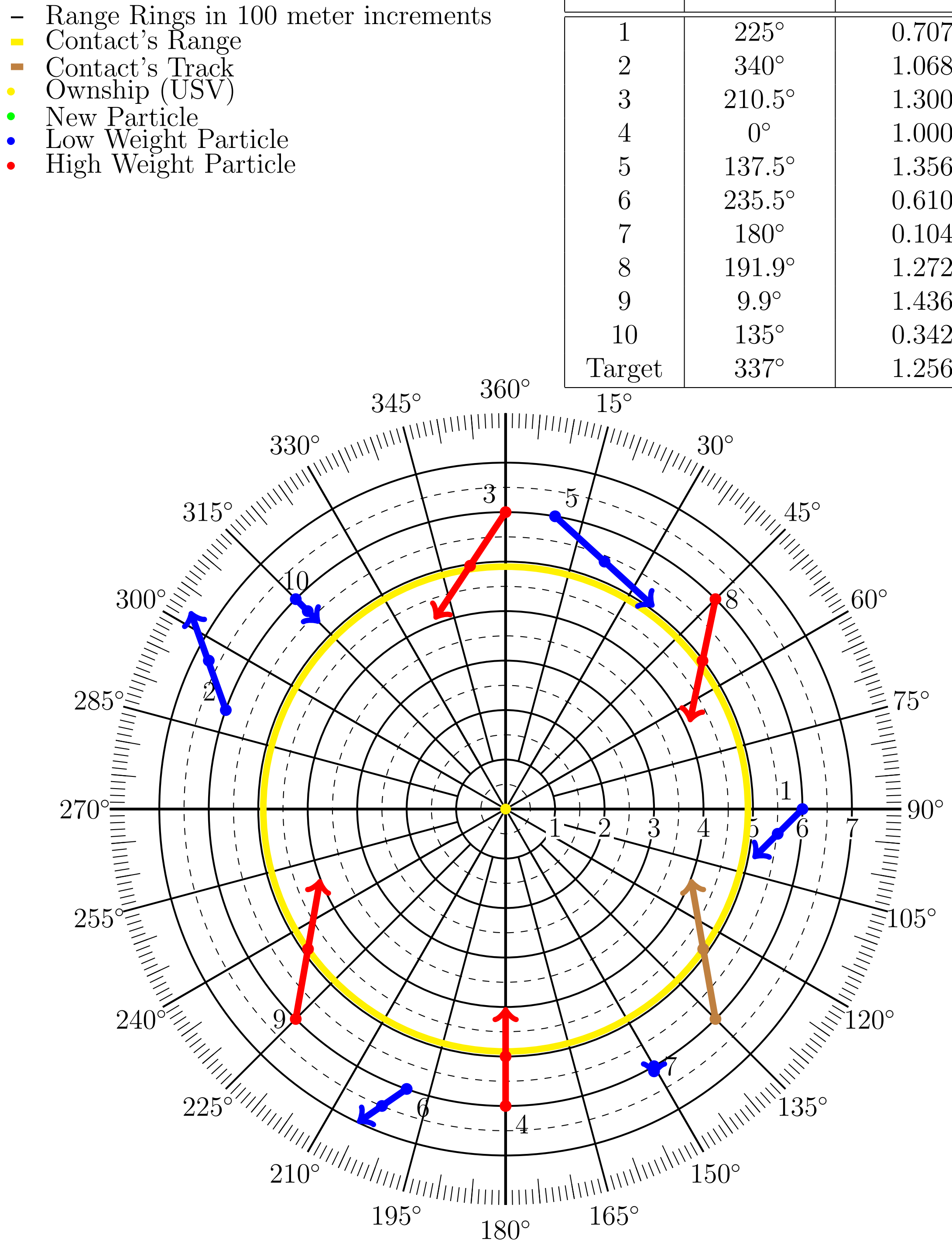


Figure 4-3: Relative Motion Plot of Particles at t = 1: After the first time step four of the ten particles have matching relative motion. Notice that each of the four particles have different starting positions, courses, and speeds.

– Range Rings in 100 meter increments
▬ Contact's Range
▬ Contact's Track
● Ownship (USV)
● New Particle
● Low Weight Particle
● High Weight Particle

Particle	Course (°)	Speed $\frac{meters}{second}$	Weight
3	210.5°	1.300	0.195
4	0°	1.000	0.073
8	191.9°	1.272	0.195
9	9.9°	1.436	0.195
Target	337°	1.256	N/A

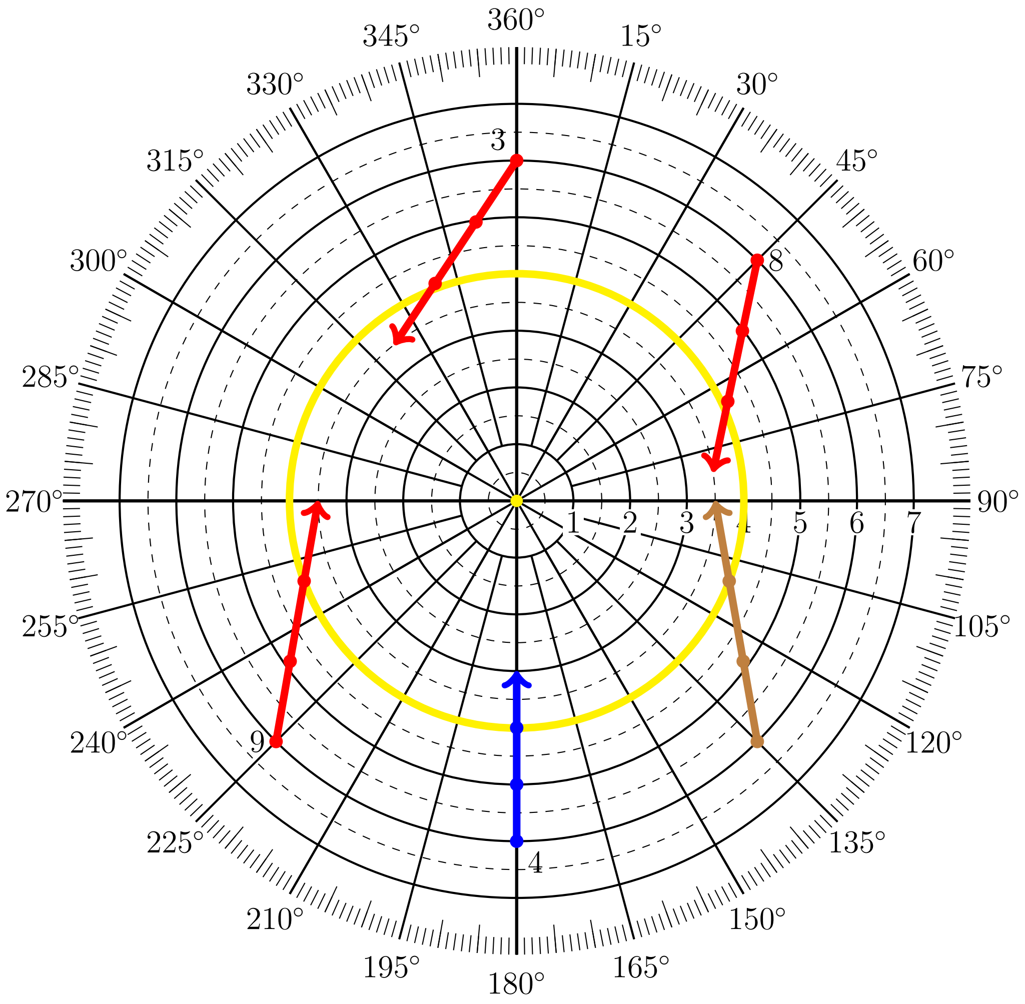


Figure 4-4: Relative Motion Plot of Particles at t = 2: As time goes on more particles can be eliminated through weight calculation; however, several particles still exist as possible hypothesis's for the contact.

– Range Rings in 100 meter increments
▬ Contact's Range
▬ Contact's Track
● Ownship (USV)
● New Particle
● Low Weight Particle
● High Weight Particle

Particle	Course (°)	Speed $\frac{meters}{second}$	Weight
3	210.5°	1.300	0.216
8	191.9°	1.272	0.216
9	9.9°	1.436	0.216
Target	337°	1.256	N/A

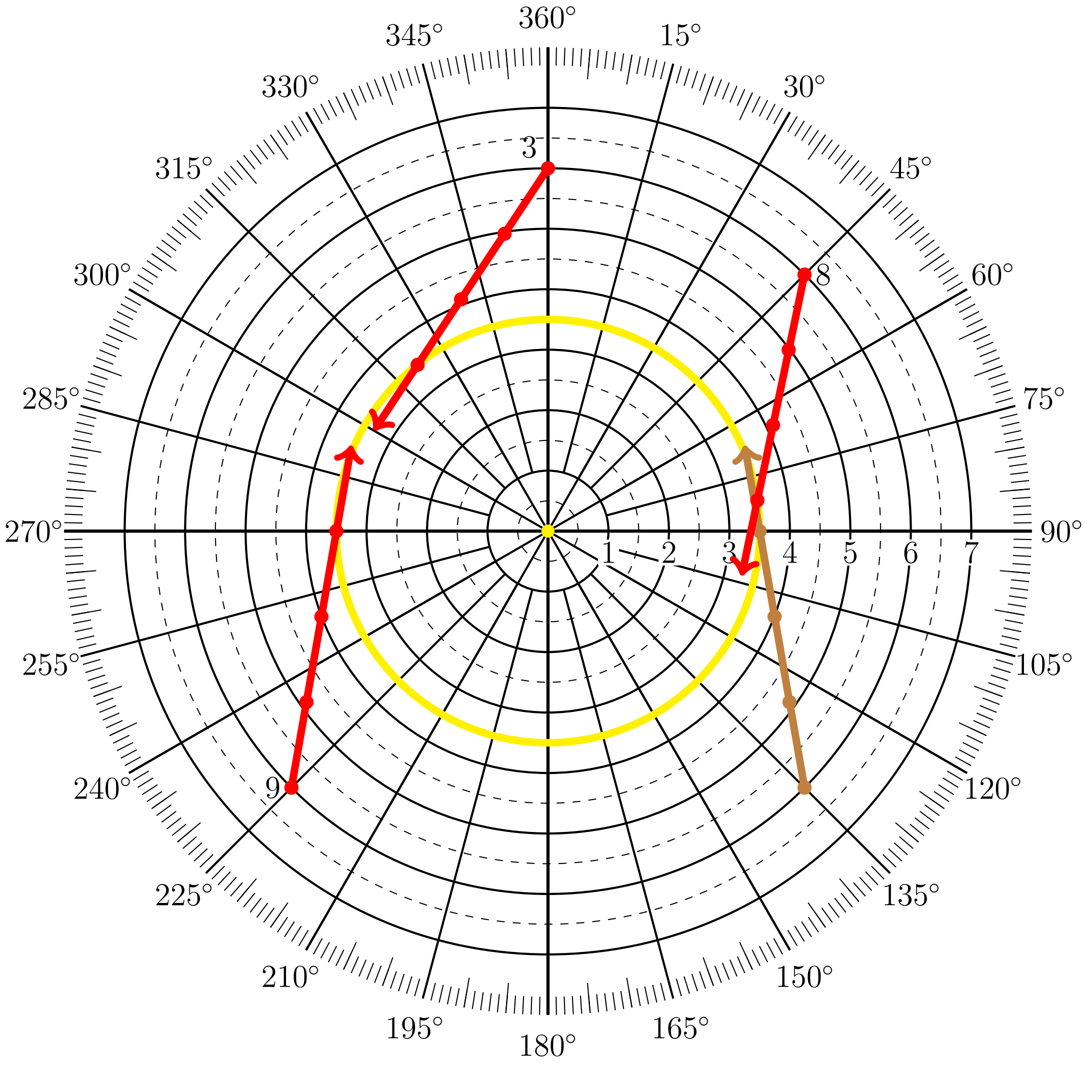


Figure 4-5: Relative Motion Plot of Particles at t = 3: Multiple hypotheses remain even after the contact transitions from a closing aspect to an open aspect. The three remain particles transition from a closing to an opening at the same time. This demonstrates a need to disambiguate the particles to find the contacts solution.

4.2 Disambiguating the Particle Filter

There are four distinct ways to provide convergence to the particle filter [31,34,39,40]:

- Know the initial state of the contact.
- Add a non-range dependent sensor (i.e. bearing) to the USV that allows for cross-correlation between sensors.
- Provide an additional range sensor from a different location and apply the new range information to the particle filter. This could be done through the collaboration of additional USVs.
- Maneuver the USV to disambiguate from the other previously feasible particles [34].

4.2.1 Knowledge of Initial State

Knowledge of the initial state can be achieved by either observing or by making assumptions in certain geographic situations. For example, it could be estimated that a contact is traveling at typical transit speed, and if that contact enters a channel or choke point an initial state can be deduced.

4.2.2 Additional Sensor

Adding another sensor like-bearing allows for the vehicle to disambiguate the particles and find the contact of concern through sensor correlation (Figure 4-6).

4.2.3 Range Correlation

Another method for particle filter correction is through fusing range information from another source. Figures 4-7 and 4-8 show how in two time steps two vehicles sharing range information can localize the contact. Figure 4-7 is the first time step. The particles are formed at the intersection of the two range circles. At the second time step Figure 4-8 one of the particle clusters was eliminated due to the distance from the

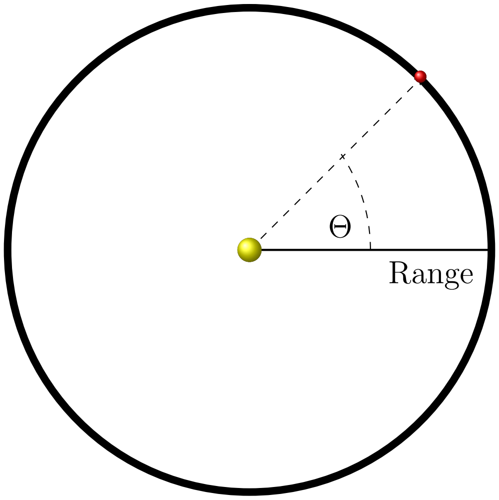
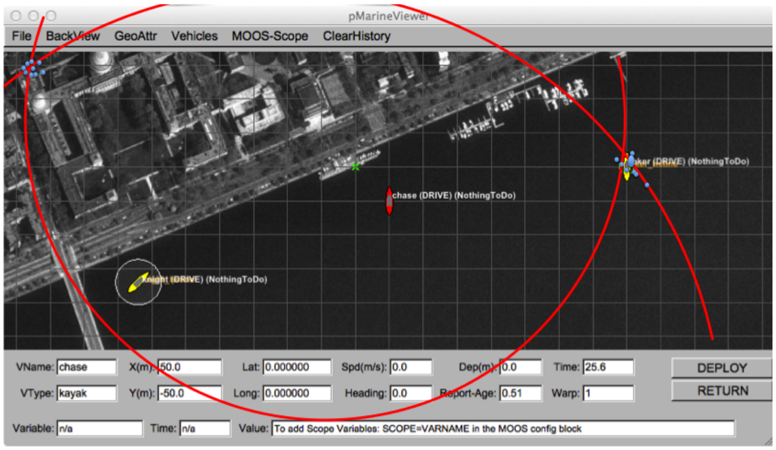


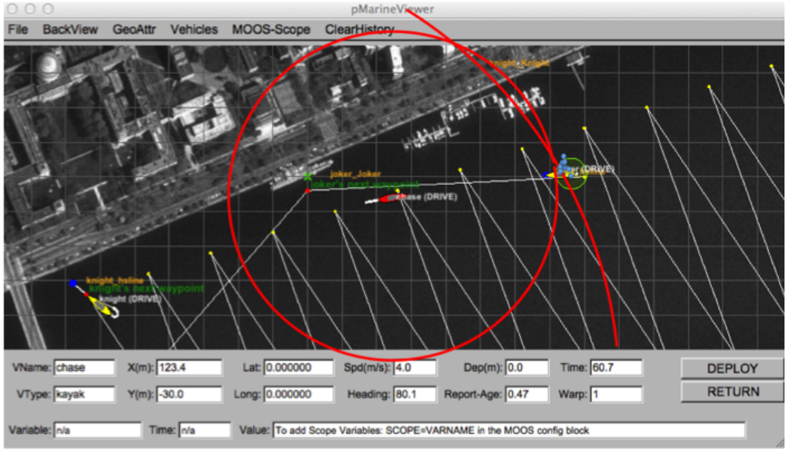
Figure 4-6: Sensor Correlation: One of the ways to remove hypothesis ambiguity is through correlated range measurements with other sensor measurements, in this case bearing. With both range and bearing information, the contact can be localized.

actual range measurements. This method demonstrates the effectiveness of multiple vehicles sharing range information and the speed at which ambiguity was removed; an additional demonstration of this is shown [42].


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
chase (DRIVE) (NothingToDo)
knight (DRIVE) (NothingToDo)
(DRIVE) (NothingToDo)
DEPLOY
RETURN

VName: chase
X(m): 50.0
Lat: 0.000000
Spd(m/s): 0.0
Dep(m): 0.0
Time: 25.6
VType: kayak
Y(m): -50.0
Long: 0.000000
Heading: 0.0
Report-Age: 0.51
Warp: 1
Variable: n/a
Time: n/a
Value: To add Scope Variables: SCOPE=VARNAME in the MOOS config block

Figure 4-7: Range Fusion at t=0: Initially the two collaborating vehicles have two clusters of particles for the contacts hypothesized position. The red circles are range circles where the particles are generated at the intersections of the shared range circles for each vehicle.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
joker_Joker
chase (DRIVE)
knight_hsline
knight's next waypoint
knight (DRIVE)
DEPLOY
RETURN

VName: chase
X(m): 123.4
Lat: 0.000000
Spd(m/s): 4.0
Dep(m): 0.0
Time: 60.7
VType: kayak
Y(m): -30.0
Long: 0.000000
Heading: 80.1
Report-Age: 0.47
Warp: 1
Variable: n/a
Time: n/a
Value: To add Scope Variables: SCOPE=VARNAME in the MOOS config block

Figure 4-8: Range Fusion at t=1: At the next time step the two collaborating ASVs were able to eliminate one of the clusters. This result is true as long as the collaborating vehicles do not have zero relative motion with one another.

4.2.4 Adaptive Maneuvering

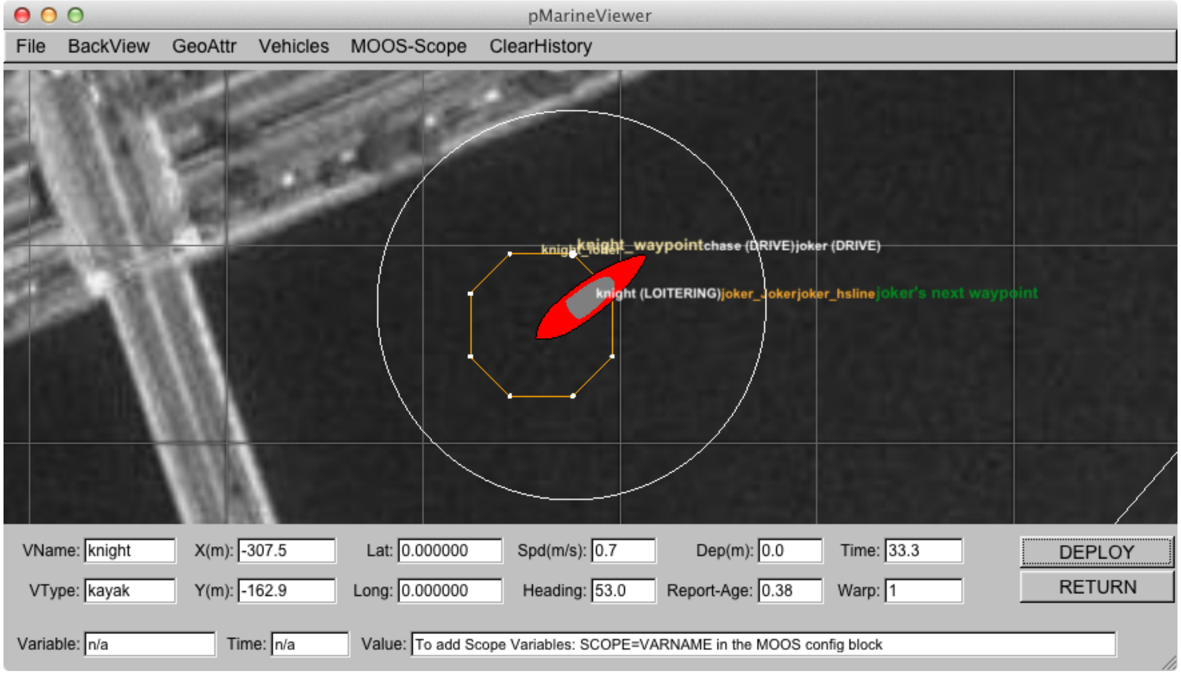
Recall that from Equations 4.7 and 4.8 that the state of range only measurements were not observable. Using Modified Polar Coordinates (MPC) and the associated transformations –in which a_{ij} are arbitrary constants where at least one constant is nonzero [34]–the following conditions for local observability were obtained:

$$\begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} \neq \begin{pmatrix} a_{11} + a_{12}\Delta t + a_{13}\Delta t^2 \\ a_{21} + a_{22}\Delta t + a_{23}\Delta t^2 \end{pmatrix} \tag{4.9}$$

Equation 4.9 implies that if the relative bearing between ownship and the contact is constant, or if ownship is traveling at a constant velocity, or ownship is traveling at a constant acceleration, the system is unobservable; however, if the contact's acceleration is zero, the Equation 4.9 reduces to the following:

$$\begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} \neq \begin{pmatrix} a_{11} + a_{12}\Delta t \\ a_{21} + a_{22}\Delta t \end{pmatrix} \tag{4.10}$$

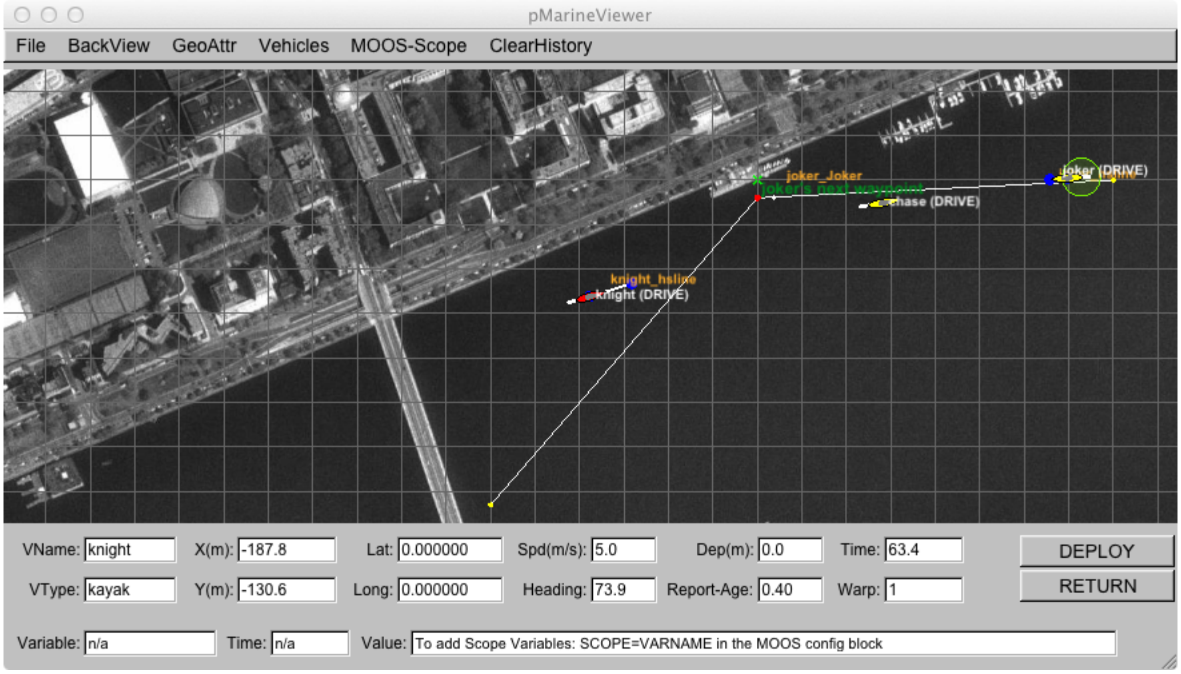
Equation's 4.10 observability requirement is violated if both ownship and the contact maintain a constant track. Equation 4.10 also requires that ownship maintain a constant velocity. From these conditions two types of adaptive maneuvers were used to disambiguate particles. The first is the initialization maneuver. The initialization maneuver performs a loiter pattern (Figure 4-9) upon detection of the contact. This ensures that Equation 4.10 becomes observable. It is important to note that this maneuver commences with the initialization of the particle filter.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
knight_loiter
knight_waypoint
chase (DRIVE)
joker (DRIVE)
knight (LOITERING)
joker_Joker
joker_hsline
joker's next waypoint
DEPLOY
RETURN

VName: knight
X(m): -307.5
Lat: 0.000000
Spd(m/s): 0.7
Dep(m): 0.0
Time: 33.3
VType: kayak
Y(m): -162.9
Long: 0.000000
Heading: 53.0
Report-Age: 0.38
Warp: 1
Variable: n/a
Time: n/a
Value: To add Scope Variables: SCOPE=VARNAME in the MOOS config block

Figure 4-9: Initial Localization Maneuver through Loitering: This is done to remove the ambiguous hypotheses by changing the relative motion between ownship and the contact. Ownship performs each turn after a series of range measurements to ensure enough information is obtained to promote differences in the relative motions of the ambiguous hypotheses.

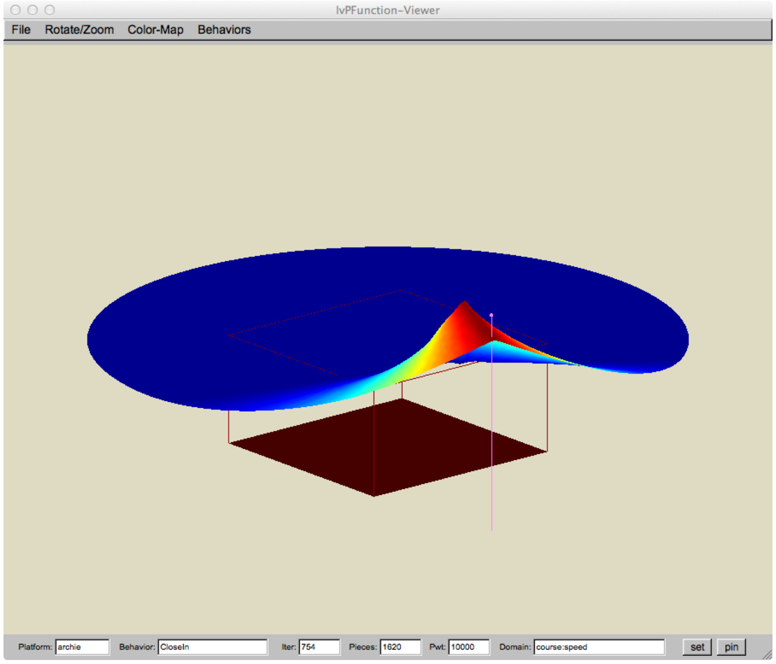
The second adaptive maneuver that was used was the intercept maneuver. The intercept maneuver (Figure 4-10) uses a built-in MOOS behavior known as $BHV\_Cutrange$. The behavior acts to reduce the range between ownship and the contact through either pursuing the contact's node report position or the Closest Point of Approach (CPA) with the contact. Because the particles are constantly fluctuating due to the introduction of noise, these course changes provide enough relative velocity to disambiguate the possible solutions that are generated through the degradation of the particles. If after a certain amount of time ownship's course remains constant, an offset is added to the particle solutions (highest and average weight) to ensure that ownship does not maintain a constant track.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
joker_Joker
joker's next waypoint
joker (DRIVE)
chase (DRIVE)
knight_hsline
knight (DRIVE)
DEPLOY
RETURN

VName: knight
X(m): -187.8
Lat: 0.000000
Spd(m/s): 5.0
Dep(m): 0.0
Time: 63.4
VType: kayak
Y(m): -130.6
Long: 0.000000
Heading: 73.9
Report-Age: 0.40
Warp: 1
Variable: n/a
Time: n/a
Value: To add Scope Variables: SCOPE=VARNAME in the MOOS config block

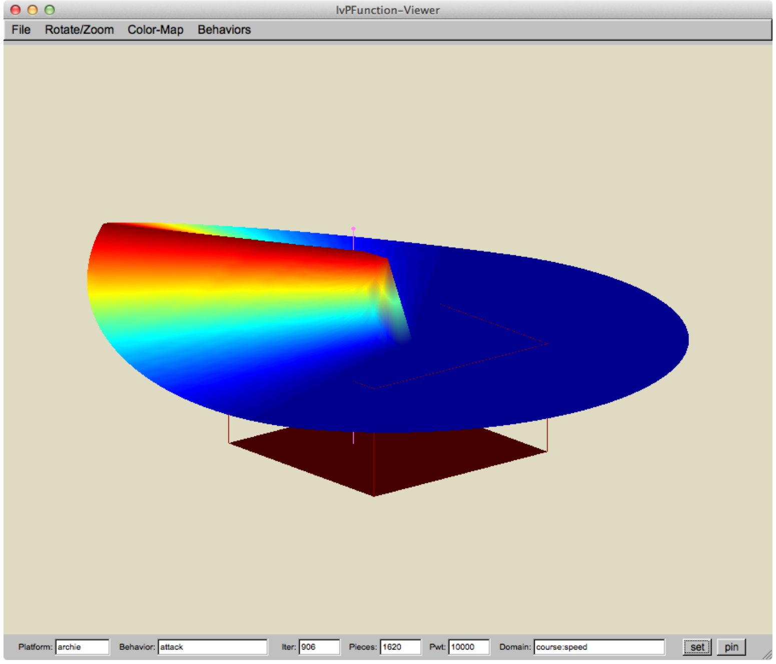
Figure 4-10: Intercepting Maneuver: Once the contact has been localized, ownship uses $BHV\_Cutrange$ to pursue two projected hypotheses, the highest weighted and the average particle position. If ownship's course has not changed in several range measurements, a 45° offset is added to the average particle position. This is done to remove the possible loss of observability from a constant relative motion.

The intercept maneuver prosecutes two different solutions. The first solution is based on the highest weighted particle (Figure 4-11), while the second solution is the average of the particles (Figure 4-12). These solutions are used in the $BHV\_Cutrange$, $HelmIvP$ (Figure 4-13) which uses these to find the most appropriate course to pursue.


IvPFunction-Viewer
File
Rotate/Zoom
Color-Map
Behaviors
set
pin

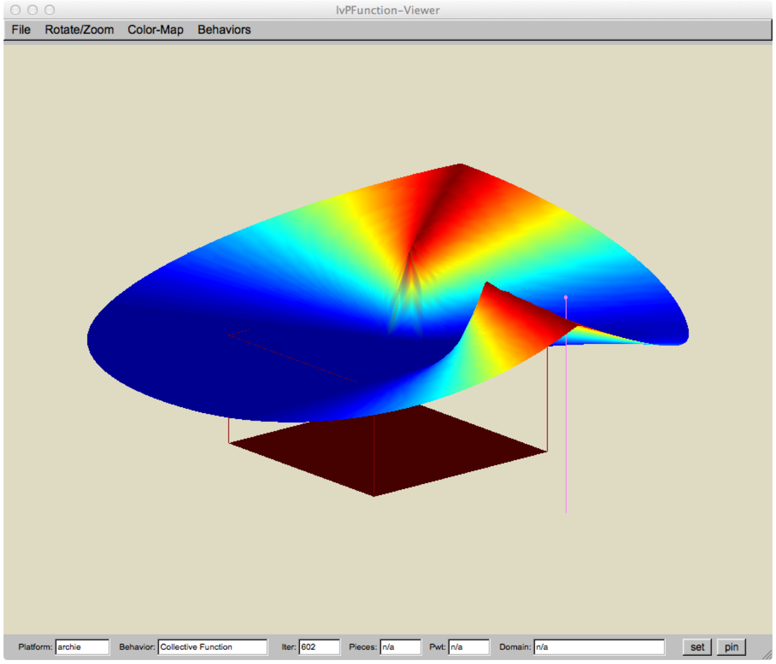
Platform: archie
Behavior: CloseIn
Iter: 754
Pieces: 1620
Pwt: 10000
Domain: course:speed

Figure 4-11: Visualization of IvPHelm using $BHV\_Cutrange$ with the Highest Weighted Particle as the target.


IvPFunction-Viewer
File
Rotate/Zoom
Color-Map
Behaviors
set
pin

Platform: archie
Behavior: attack
Iter: 906
Pieces: 1620
Pwt: 10000
Domain: course:speed

Figure 4-12: Visualization of IvPHelm using $BHV\_Cutrange$ with the Average Weighted Particle Position as the target.


IvPFunction-Viewer
File
Rotate/Zoom
Color-Map
Behaviors
set
pin

Platform: archie
Behavior: Collective Function
Iter: 602
Pieces: n/a
Pwt: n/a
Domain: n/a

Figure 4-13: Visualization of IvPHelm using the collective of $BHV\_Cutrange$ behaviors.

Chapter 5

Simulations, Testing, and Analysis

This chapter discusses some of the particle filter settings and their effectiveness on solution convergence, the simulations that were conducted, and the in water Charles River test. The results show in simple non-maneuvering or slightly maneuvering cases that the method developed for tracking was capable to tracking a submerged contact. As the contact develops more complicated patterns, tracking because more difficult requiring concepts like reserve particles to be developed and implemented.

5.1 Particle Filter Settings

5.1.1 Measures of Effectiveness

To determine the effectiveness of the particle filter, metrics were established to calculate the distance between the particles and the true position of the static contact. For this, three methods were used to estimate the position of the contact used. [1,37]

1.) The first method was to estimate the position of the contact by using the particle with the highest weight (the green triangle).

2.) The second method was to find the average position of the particles. A straight average is used even though it was not a good initial estimate (Figure 5-1). Once the initialization is complete, the group of particles centralize to one cluster and the average becomes a good estimate. Initially the average weight (the red circle) is located in the middle of the particle spread, not neighboring any particles due to the initial formation of particles along the range circle.

3.) The third method was to determine the weighted average of the particles. Testing showed this method was the most accurate but had results very similar to the average of the particles.

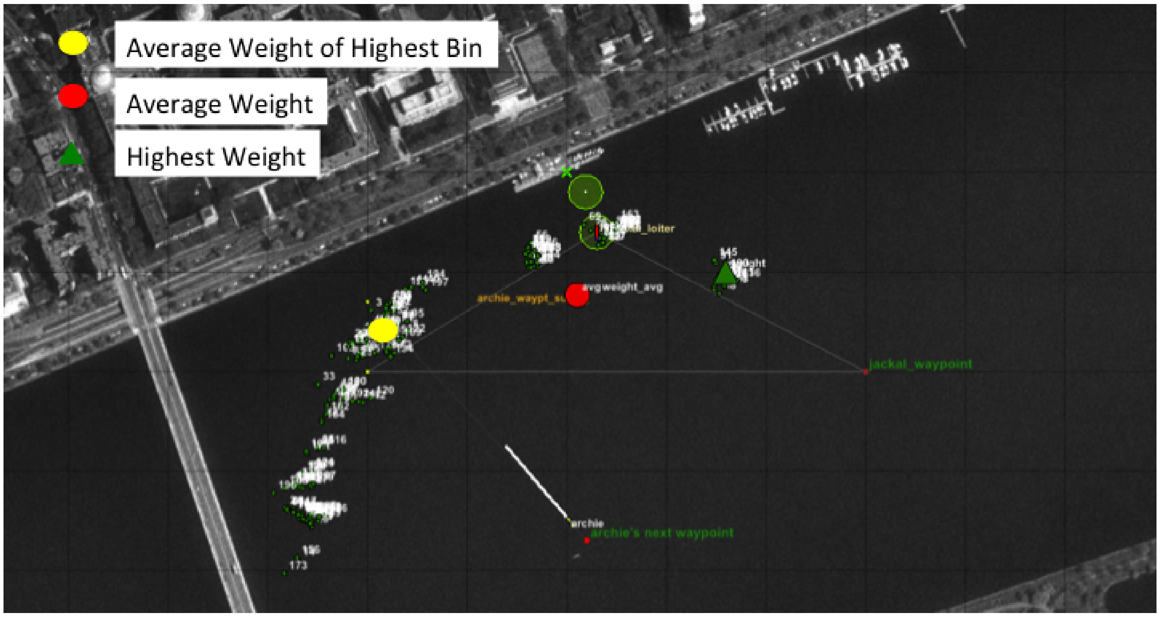

Average Weight of Highest Bin
Average Weight
Highest Weight
archie_waypt_su
avgweight_avg
jackal_waypoint
archie
archie's next waypoint

Figure 5-1: Initial Particle Separation for a Stationary Contact. Several ambiguous clusters are formed after the resample step. It is important to note that the average weight wast used as a metric despite its divergence at initialization. The average weight of the most populated cluster is shown in yellow, after a few time steps the average weight and the average weight of the most populated cluster are the same.

5.1.2 Number of Particles

One question that arises from the use of a particle filter is: How many particles are necessary in order to find a solution in a reasonable amount of time. If the filter uses a larger number of particles, the time needed to find a reliable solution is decreased but the overall computational power that was expended is increased. Conversely, the fewer number of particles used, the more time required to find a solution. To evaluate this trade off, a series of simulations was run to observe the trade space between the number of particles and the time required for solution convergence.

Establishing the resolution needed for the solution and the range of the key variables are also vital aspects for determining the number of particles required for convergence. This implementation of the particle filter contains the following variables and their associated ranges (Table 5.1).

Variable	Range
Max Depth	0-30 meters
Bearing	0-360°
Course	0-360°
Speed	0-10$\frac{meters}{second}$

Table 5.1: Table of Variables

Figure 5-2 shows the number of time steps needed for convergence versus the number of particles used. Several trials were conducted with fewer than 50 particles and all failed to converge. For the purposes of this thesis, 2,500 particles were used for all of the follow-up scenarios.

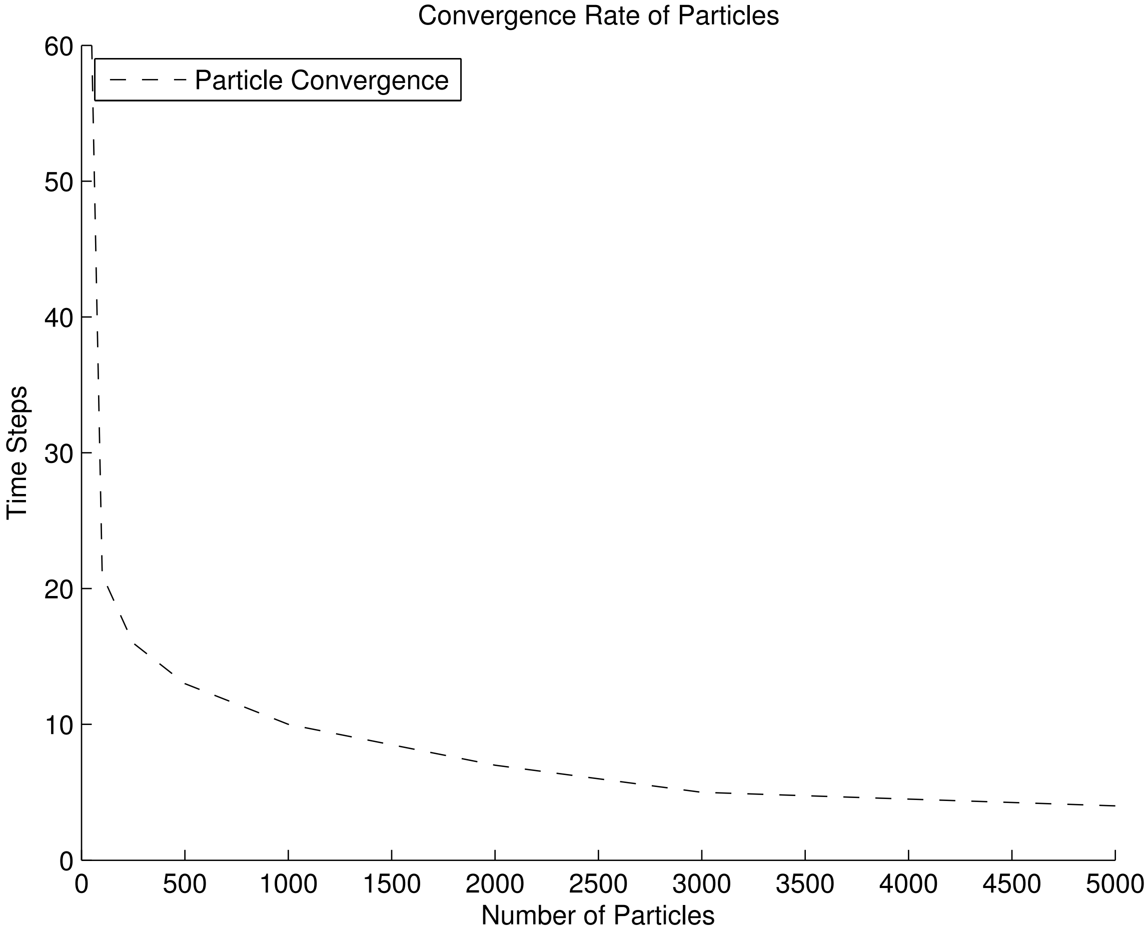
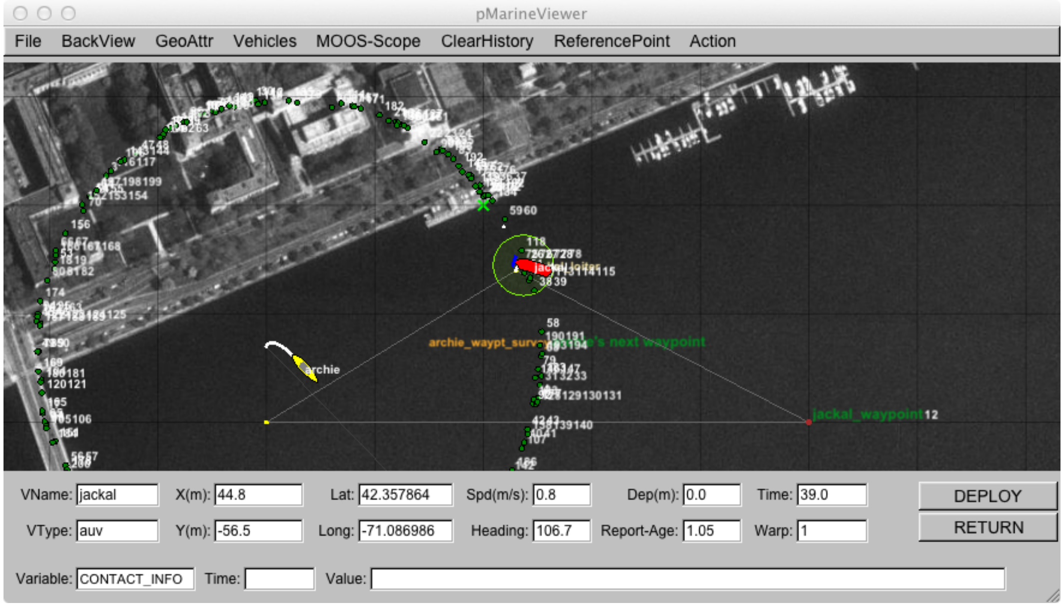


Figure 5-2: Range Convergence for the number of particles over time. At some point increasing the number of particles does not significantly improve the rate of convergence. 2500-3000 particles appear to be the optimal number of particles for this project.

5.1.3 Moving Contact Settings

The next step was to determine the effectiveness of the particle filter for a moving contact. For this, the ASV and the adversary drove past each other, maintaining their respective courses and speeds as indicated by Figure 5-3 through Figure 5-7. The red vessel is the contact of interest, the yellow vessel is ownship, and the green balls represent the generated particles. Initially, the particles are formed radially from the first range measurement; as time progresses, the particles coalesce into larger groupings. At t=5 (Figure 5-7) the particles have converged on the contact.

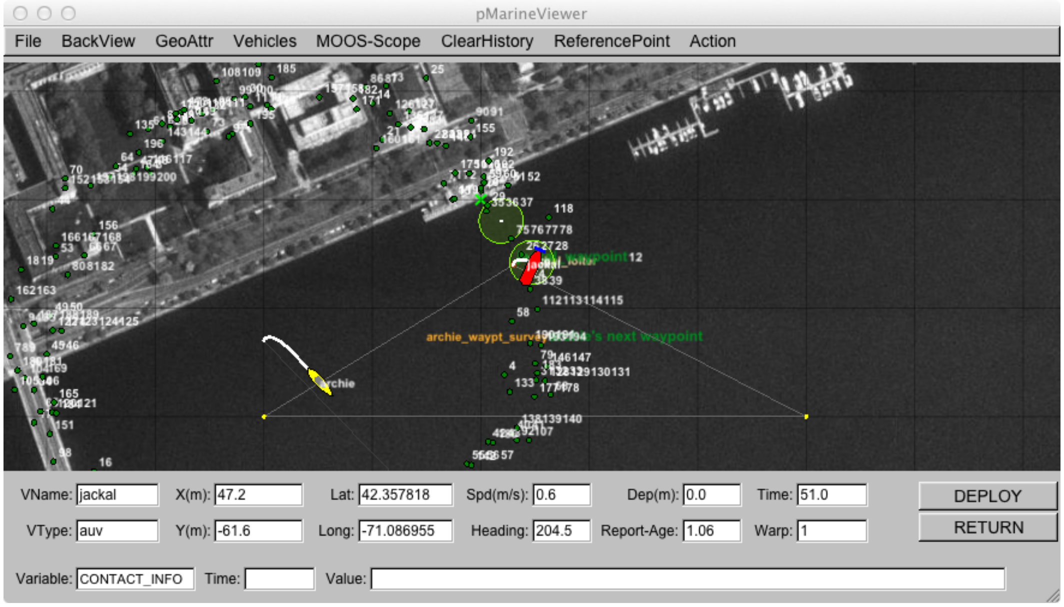

pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
ReferencePoint
Action
archie
archie_waypt_surv
jackal_waypoint
DEPLOY
RETURN

VName: jackal X(m): 44.8 Lat: 42.357864 Spd(m/s): 0.8 Dep(m): 0.0 Time: 39.0

VType: auv Y(m): -56.5 Long: -71.086986 Heading: 106.7 Report-Age: 1.05 Warp: 1

Variable: CONTACT_INFO Time: Value:

Figure 5-3: Moving Target at t=1: Once ownship received range information, initialization of particles occur along that radius.

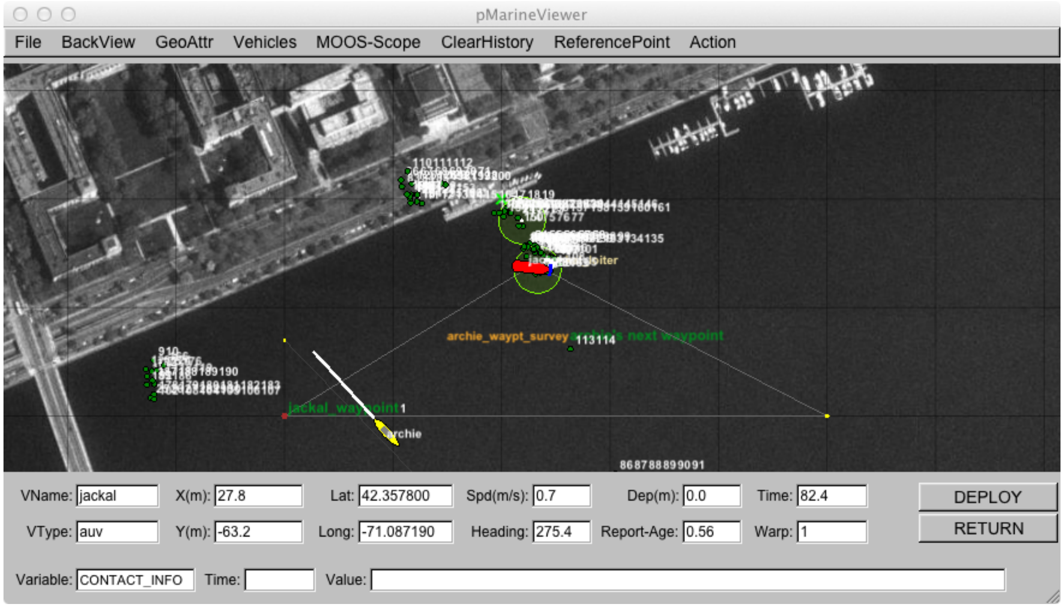

pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
ReferencePoint
Action
archie
archie_waypt_surv
DEPLOY
RETURN

VName: jackal X(m): 47.2 Lat: 42.357818 Spd(m/s): 0.6 Dep(m): 0.0 Time: 51.0

VType: auv Y(m): -61.6 Long: -71.086955 Heading: 204.5 Report-Age: 1.06 Warp: 1

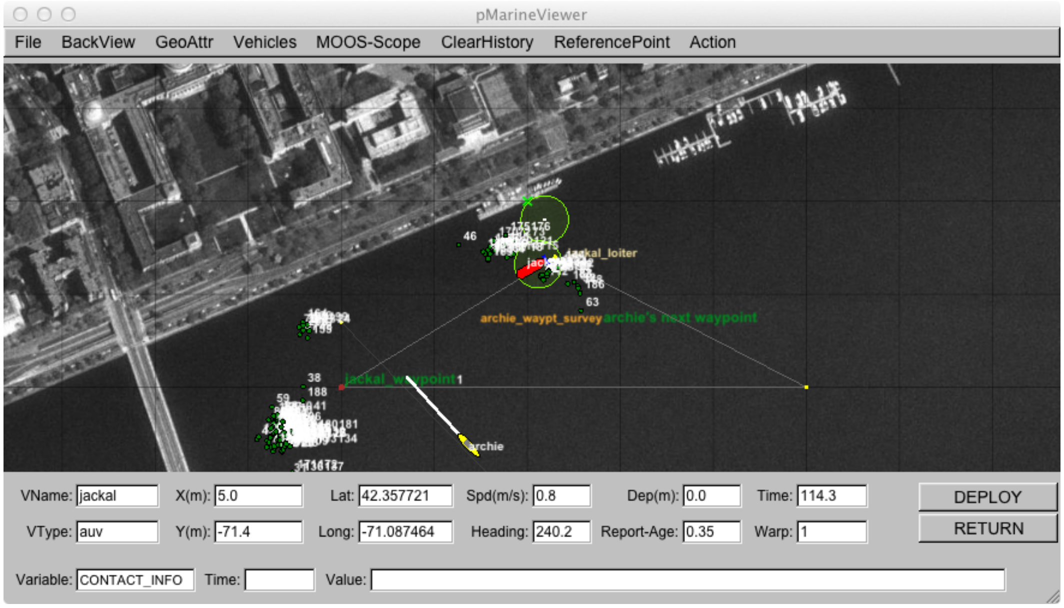
Variable: CONTACT_INFO Time: Value:

Figure 5-4: Moving Target at t=2: Particle separation begins as a result of each particle being initialized with different courses and different speeds.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
ReferencePoint
Action
jackal_loiter
archie_waypt_survey
archie's next waypoint
jackal_waypoint1
archie
DEPLOY
RETURN

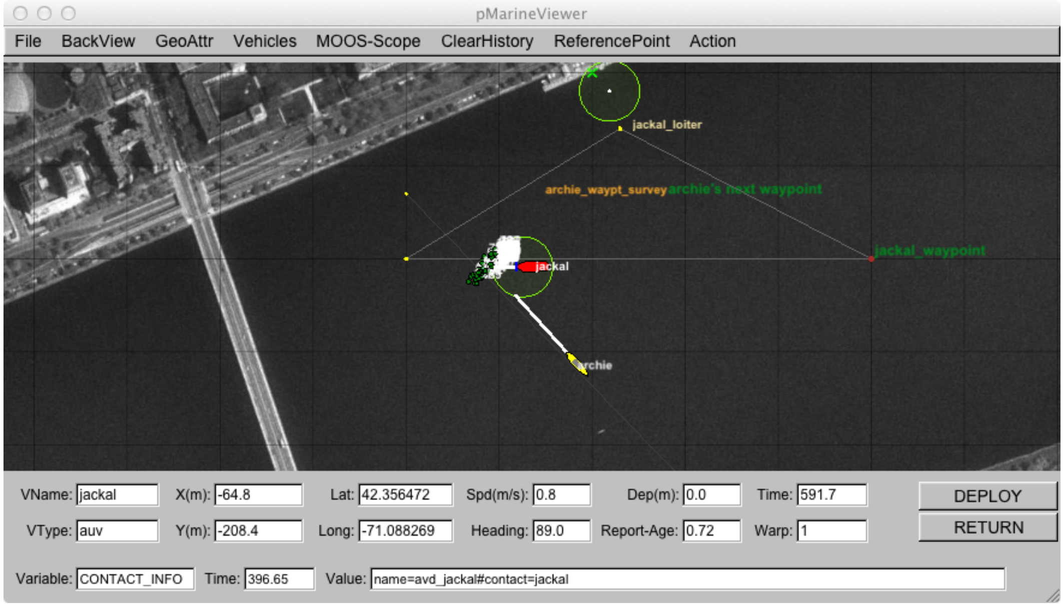
VName: jackal
X(m): 27.8
Lat: 42.357800
Spd(m/s): 0.7
Dep(m): 0.0
Time: 82.4
VType: auv
Y(m): -63.2
Long: -71.087190
Heading: 275.4
Report-Age: 0.56
Warp: 1
Variable: CONTACT_INFO
Time:
Value:

Figure 5-5: Moving Target at t=3: Particles begin to form cluster with similar relative motion as a result of the resampling subroutine transforming the particle distribution by drawing with replacement from the previous distribution.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
ReferencePoint
Action
jackal_loiter
archie_waypt_survey
archie's next waypoint
jackal_waypoint1
archie
DEPLOY
RETURN

VName: jackal
X(m): 5.0
Lat: 42.357721
Spd(m/s): 0.8
Dep(m): 0.0
Time: 114.3
VType: auv
Y(m): -71.4
Long: -71.087464
Heading: 240.2
Report-Age: 0.35
Warp: 1
Variable: CONTACT_INFO
Time:
Value:

Figure 5-6: Moving Target at t=4: Clusters continue to consolidate after additional resampling events.


pMarineViewer
File
BackView
GeoAttr
Vehicles
MOOS-Scope
ClearHistory
ReferencePoint
Action
jackal_loiter
archie_waypt_survey
archie's next waypoint
jackal_waypoint
jackal
archie
DEPLOY
RETURN

VName: jackal
X(m): -64.8
Lat: 42.356472
Spd(m/s): 0.8
Dep(m): 0.0
Time: 591.7
VType: auv
Y(m): -208.4
Long: -71.088269
Heading: 89.0
Report-Age: 0.72
Warp: 1
Variable: CONTACT_INFO
Time: 396.65
Value: name=avd_jackal#contact=jackal

Figure 5-7: Moving Target at t=5: The cluster of particles eventually consolidate to the contact's location.

The Gaussian Variance Setting

Variance	Average Solution Range Error (m)	Highest Weighted Particle Solution Range Error (m)
5	48.11	41.36
15	32.25	36.07
25	40.64	42.23
50	44.54	42.12

Table 5.2: Gaussian Variance Error

Recall from Section 2.5 that the weights were determined from the probability distribution of the range difference between the predicted range and the sensed range. This section shows the results of a series of trials used to find a reliable setting for the Gaussian variance. If the Gaussian variance is set too high, then the cloud of particles will be too large, making it difficult to localize a solution. If the Gaussian Variance is set too low then correct particles could be mistakenly discarded due to natural noise. Table 5.2 shows the results for different variance settings; based on the results, a variance of 15 appears to produce the smallest average solution range error for these simulated conditions.

The Course / Speed Noise Setting

One way to promote particle variation is through the introduction of noise into both course and speed every time a particle's position is updated. One question that arises is: How much noise is necessary to achieve proper particle diversity? If too little noise is added, then particle variation is small and an accurate solution is not achieved. If too much noise is added, an excess of particle diversity is created and the solution for the contact of interest has little fidelity. Figures 5-8 and 5-9 and Table 5.4 show the effects of various noise levels on solution convergence. Note that with no noise the particles have difficulty converging on a solution, and with high noise the particles' range error is increased. It is also evident from these figures that the error is significantly reduced after time step 8; this was due to a maneuver from ownship that caused a reduction in ambiguity.

Noise Level	Range Error Before Maneuver (m)	Range Error After Maneuver (m)
5°	53.03	7.69
15°	9.87	12.02
50°	6.83	5.96
90°	8.68	5.61
180°	8.65	8.56

Table 5.3: Average Course Noise Error

Noise Level	Range Error Before Maneuver (m)	Range Error After Maneuver (m)
0.5 m/s	10.47	5.534
1 m/s	41.98	5.45
1.5 m/s	46.87	7.172
2 m/s	49.02	10.46

Table 5.4: Average Speed Noise Error

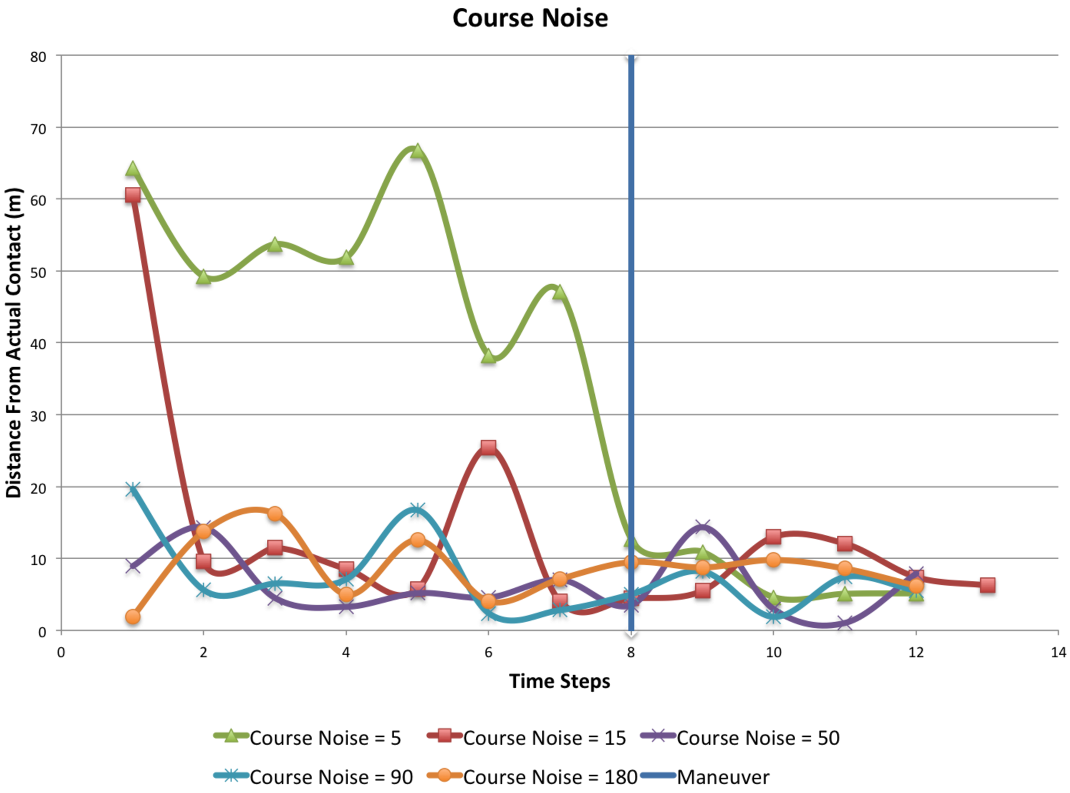


Figure 5-8: Course Noise: After each resample step the particles gradually consolidate to one particle. No significant improvements for localization were seen after a course noise of $15°$ was added. An important observation occurs at time step 7-8; here ownship maneuvered and was able to localize the contact.

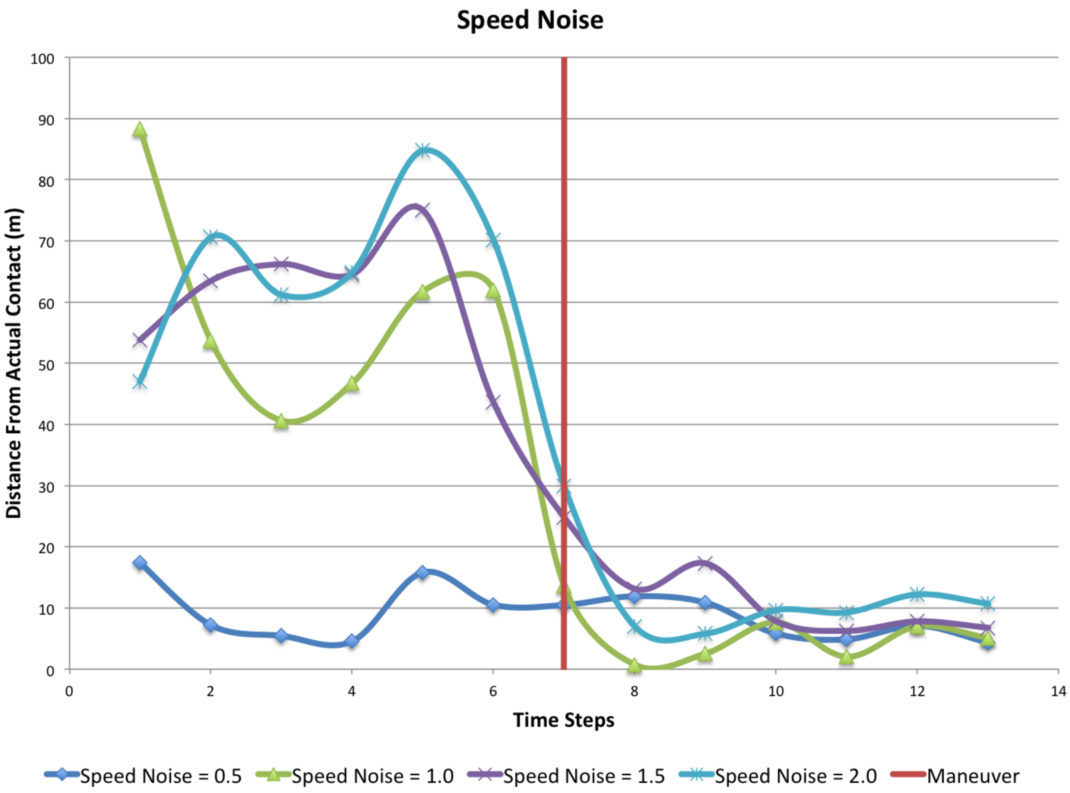


Figure 5-9: Speed Noise: After each resample step the particles gradually consolidate to one particle. No significant improvements for localization were seen after a speed noise of $0.5m/s$ was added. An important observation occurs at time step 7-8; here ownship maneuvered and was able to localize the contact.

5.2 Simulation Testing

This section illustrates a series of scenarios that tested the adaptive maneuvering and effectiveness of the particle filter. The location for the simulation and real testing was the MIT Sailing Pavilion as seen in Figure 5-10.

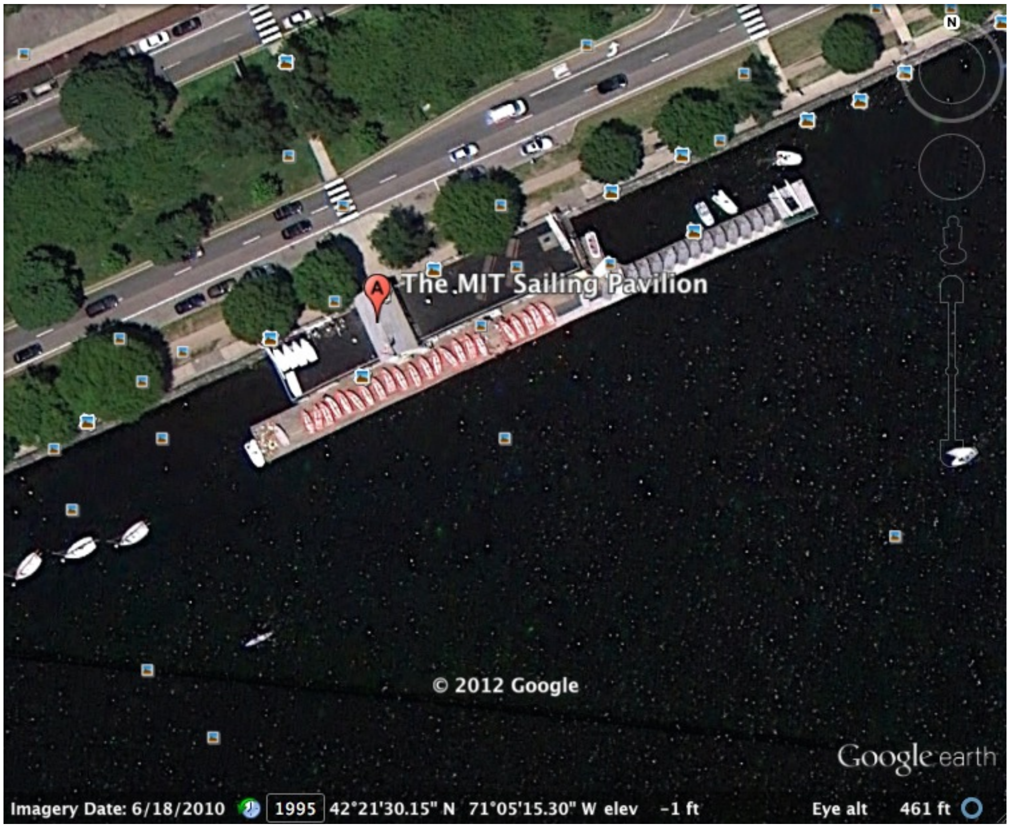

The MIT Sailing Pavilion
© 2012 Google
Google earth
Imagery Date: 6/18/2010 1995 42°21'30.15" N 71°05'15.30" W elev -1 ft Eye alt 461 ft

Figure 5-10: MIT Sailing Pavilion: Image was taken from Google Earth.

5.2.1 Box Maneuver

Figure 5-11 is an example of a contact performing a basic box pattern while using reserve particles. The average particle filter error for the maneuvering contact was 7.64 meters for the average hypothesis and 12.64 meters for the highest weighted hypothesis.

Figure 5-12 is an example of a contact performing a basic box pattern without using reserve particles. The average particle filter error for the maneuvering contact was 9.93 meters for the highest weighted hypothesis and 14.7 meters for the average hypothesis. Notice there is a slight benefit for using reserve particles.

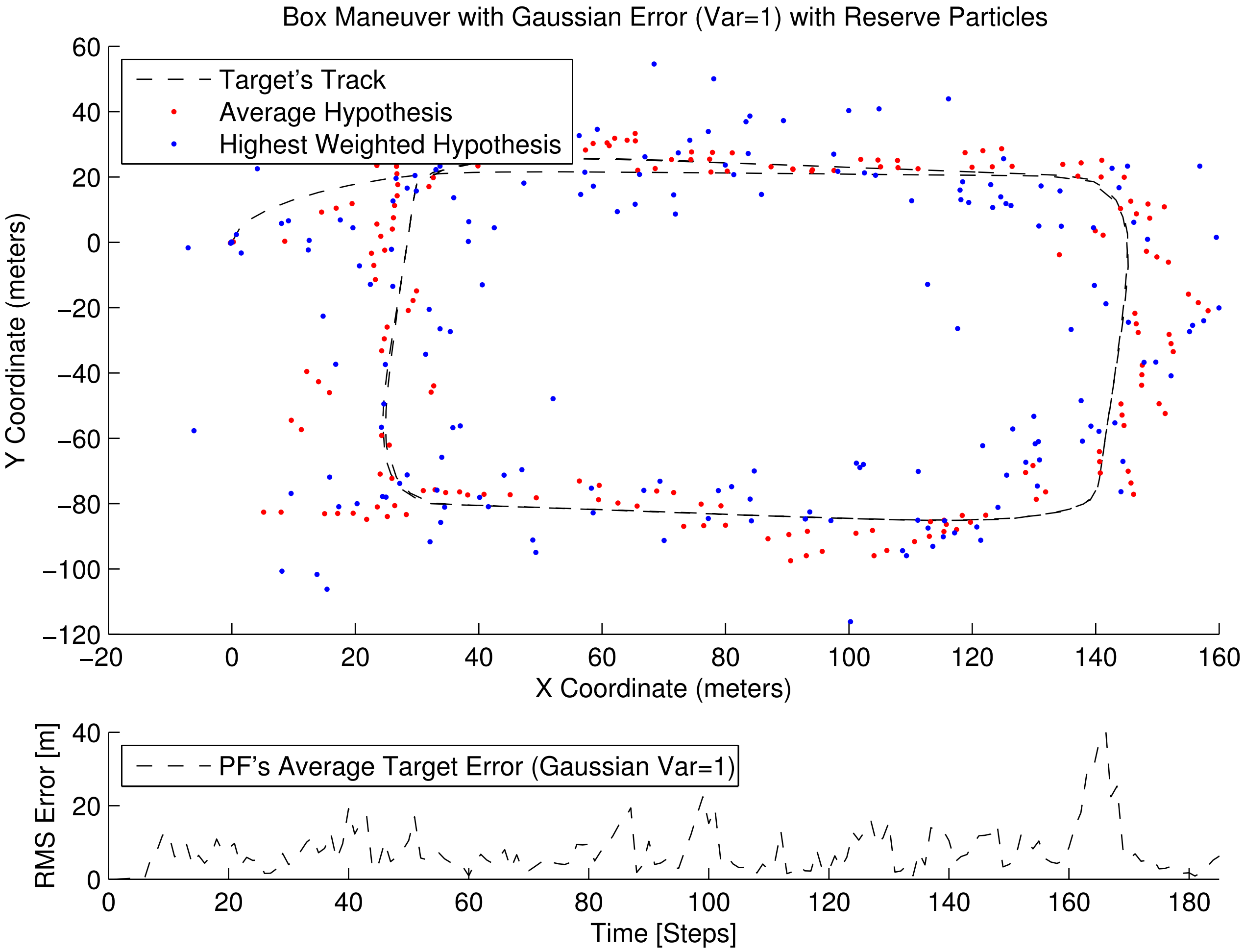


Figure 5-11: Box Simulation: Ownship was able to track a maneuvering contact within 10 meters of its position. One of the difficulties with this scenario was the contact maneuvered every 100 meters (60 seconds).

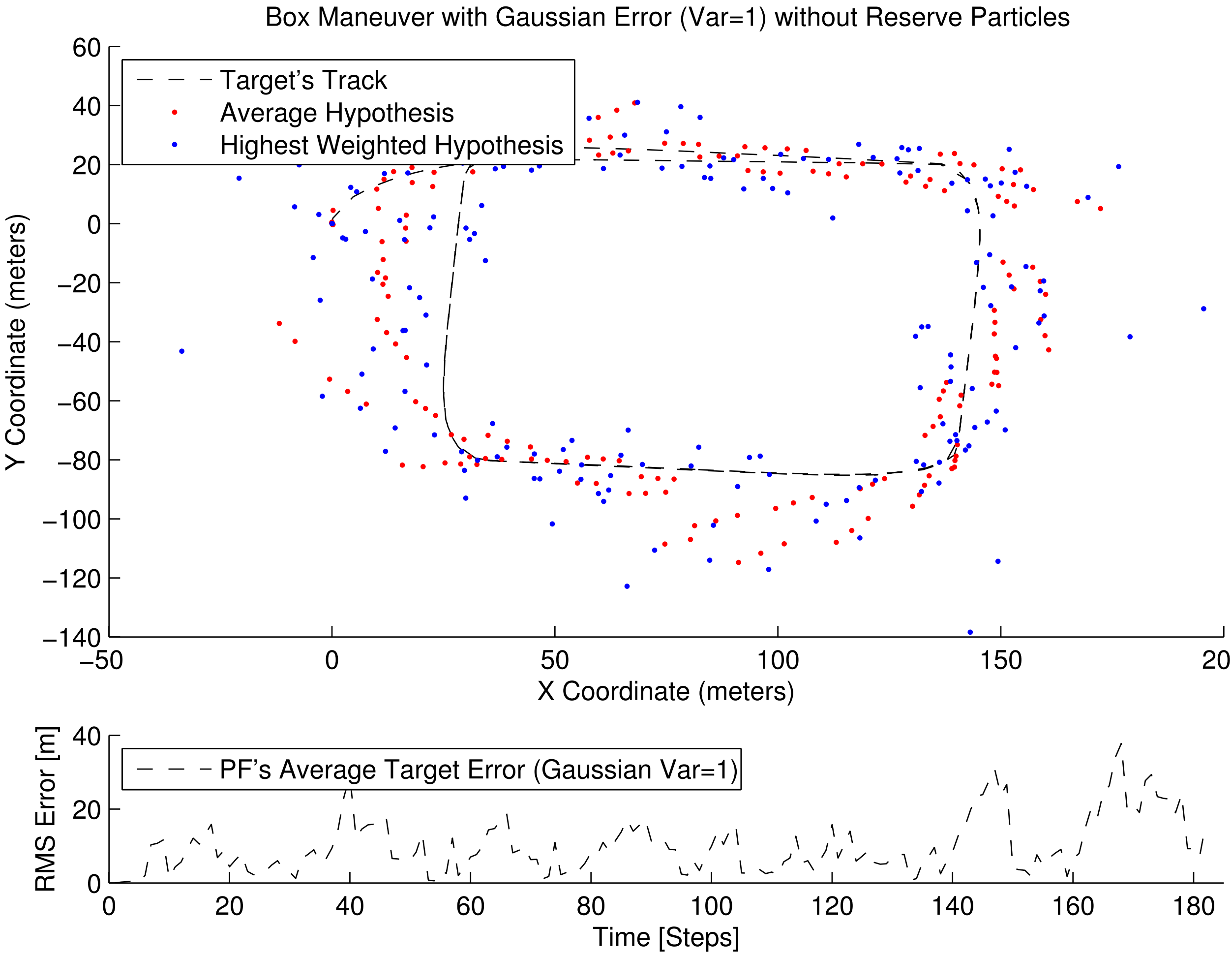


Figure 5-12: Box Simulation: Ownship was able to track a maneuvering contact within 12 meters of its position. One of the difficulties with this scenario was the contact maneuvered every 100 meters (60 seconds).

5.2.2 Bowtie Maneuver

Figure 5-13 is an example of a contact performing a basic bowtie pattern. The average particle filter error for the maneuvering contact was 9.5 meters for the highest weighted hypothesis and 12.2 meters for the average hypothesis.

Without reserve particles the algorithm could not localize and track the contact. The average particle filter error for the maneuvering contact was 17.5 meters and 19.6 meters for the highest weighted particle. However, in some trials, ownship lost complete track of the contact especially as range sampling decreased. Figure 5-14 is an great example of the particle filter being unable to match the contacts course change.

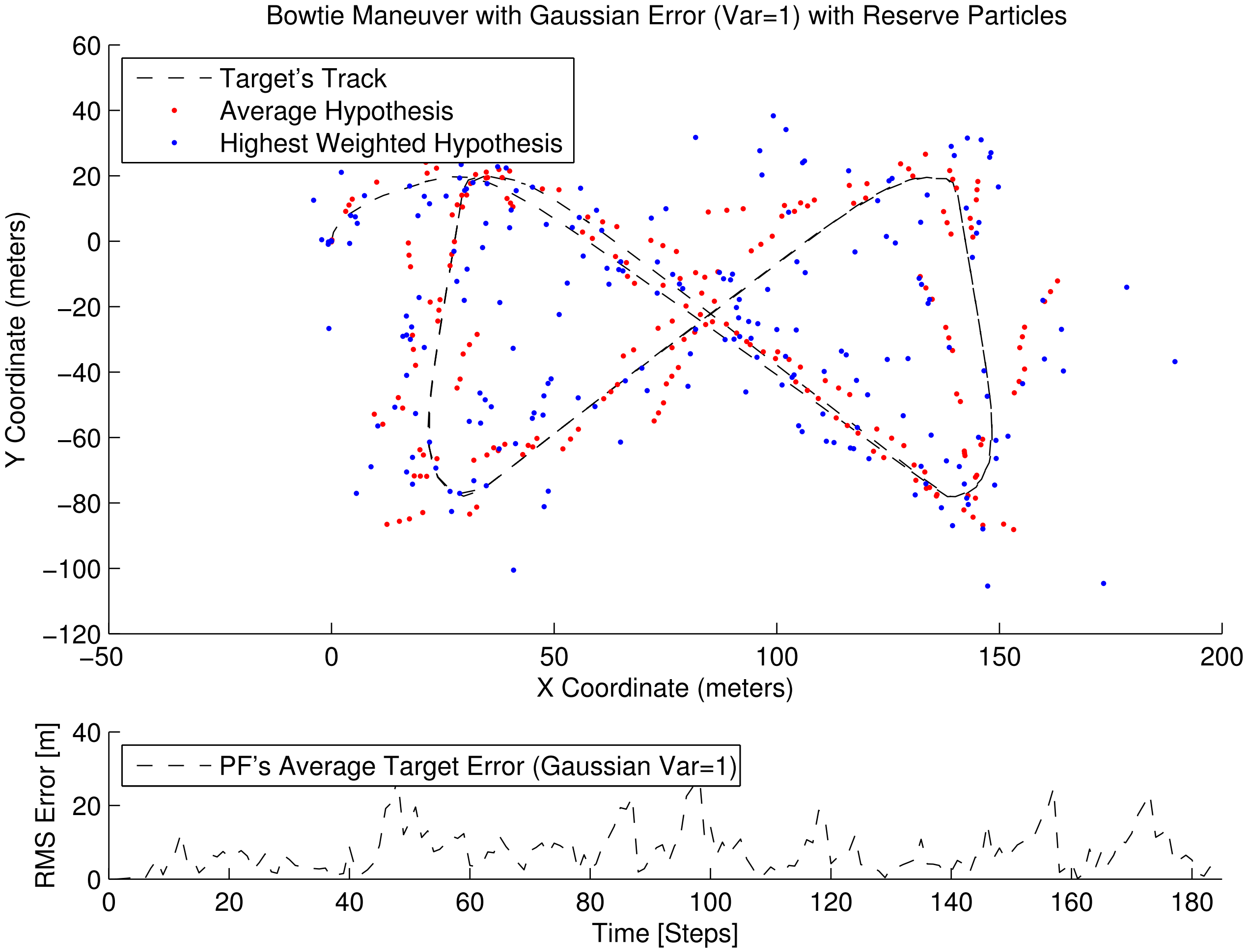


Figure 5-13: Bowtie Simulation with Reserve Particles: Ownship was able to track the contact within 12 meters. The bowtie simulation was more difficult for the algorithm due to the large course changes that occurred. Using reserve particles provided a significant reduction of error.

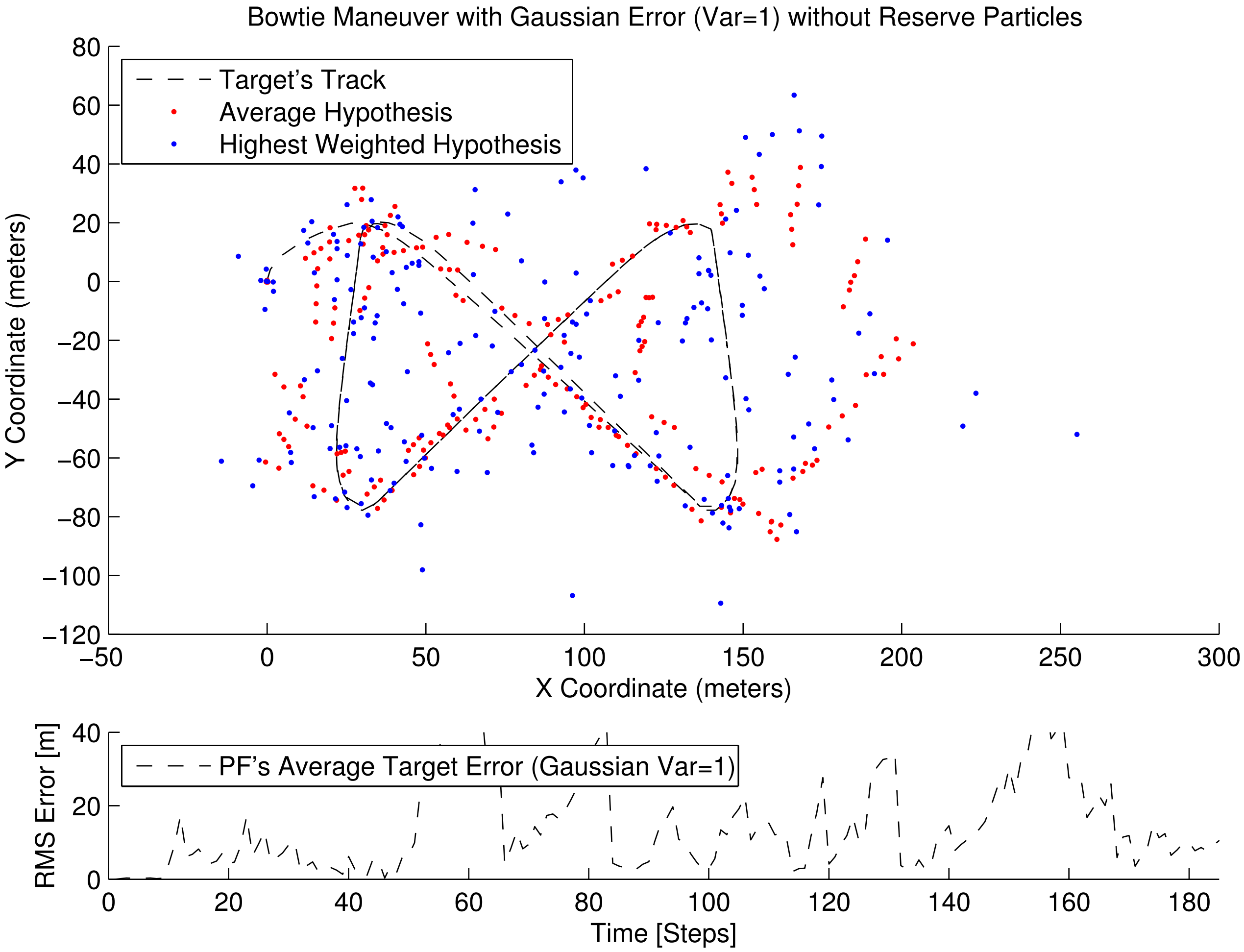


Figure 5-14: Bowtie Simulation with no Reserve Particles: The approximate error was above 20 meters and eventually lost track of the contact. The particle filter failed to sufficiently converge without the use of reserve particles.

Figure 5-15 shows two separate trials with different noise distributions, a uniform, and a gaussian. These distributions had similar results; however, one noticeable difference was the uniform noise appeared to have higher deviations when the contact maneuvered.

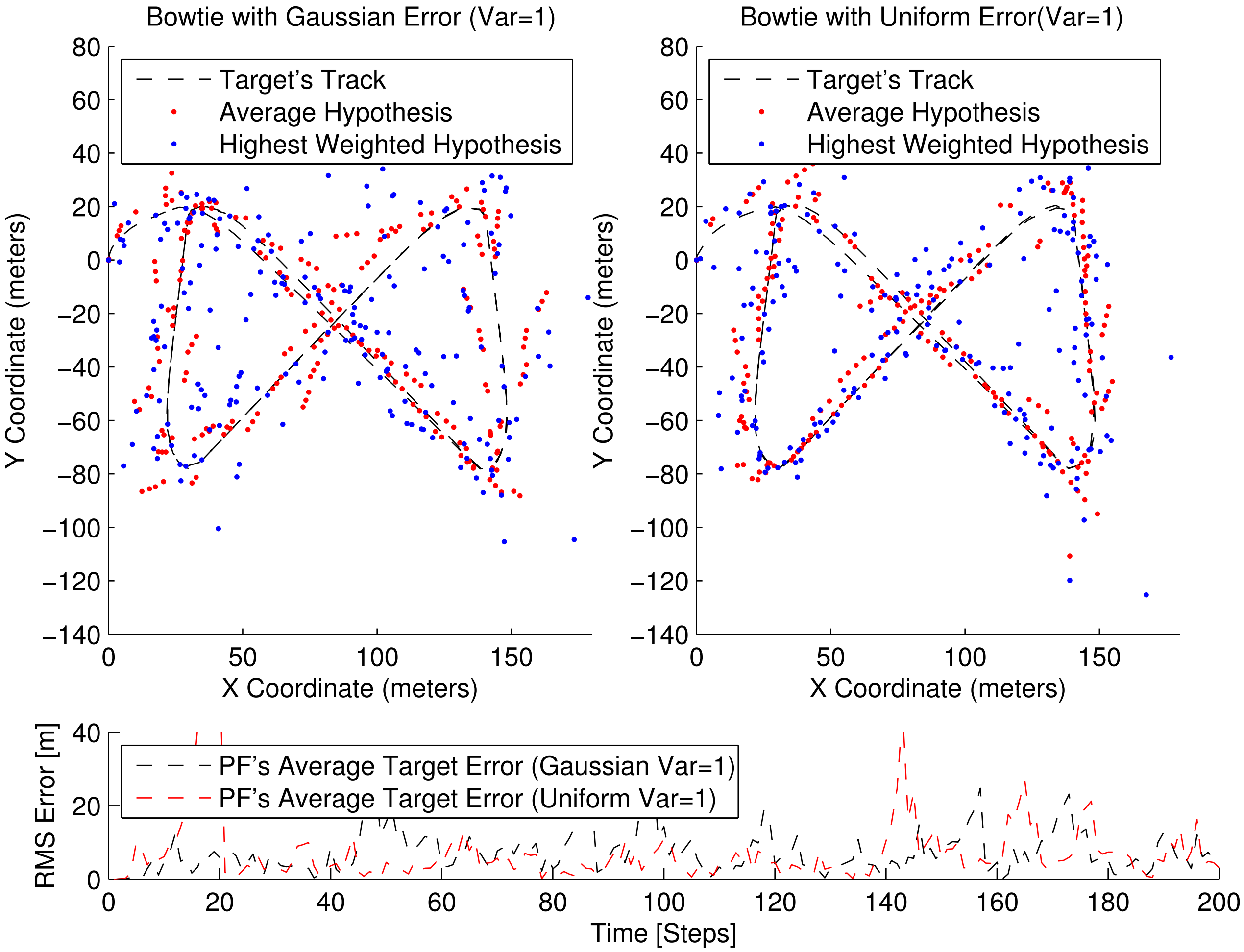


Figure 5-15: Bowtie Simulation with Reserve Particles for Uniform and Gaussian Distribution: Tracking from a source with a Gaussian error or a Uniform error appeared to be similar. One noticeable difference was the uniform noise appeared to have higher deviations when the contact maneuvered.
Ownship was able to track the contact within 12 meters. The bowtie simulation was more difficult for the algorithm due to the large course changes that occurred. Using reserve particles provided a significant reduction of error.

5.2.3 Loiter Maneuver

Figure 5-16 is an example of a contact performing a basic loitering pattern. The difficulty with this pattern is the number of maneuvers being performed by the contact. It is clear that the particle filter has less difficulty tracking this contact due to the small angle course changes. The average particle filter error for the maneuvering contact was 6.34 meters for the average hypothesis and 13.2 meters for the highest weighted hypothesis. Figure 5-17 is an example of the same loiter pattern with a 300 meter radius vice a 100 meter. The accuracy is significantly improved to an average particle filter error of 5.18 and the highest weight hypothesis error of 11.5 because of the longer time on the track segment.

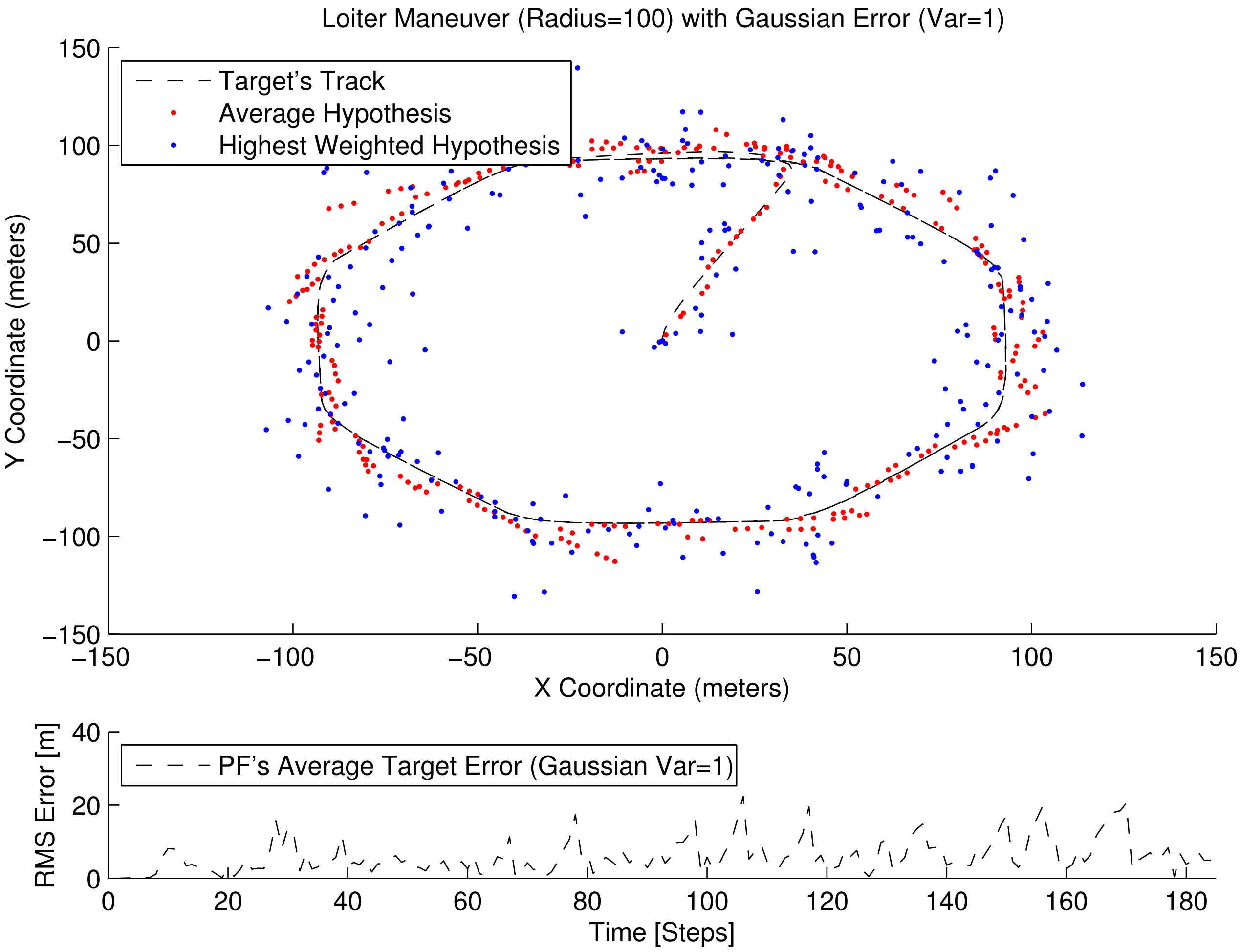


Figure 5-16: Loitering Simulation: Ownship was able to continually track the contact with the particle filter with an approximate error of 12 m. The particle filter tracks with greater accuracy if the contact performs smaller course changes.

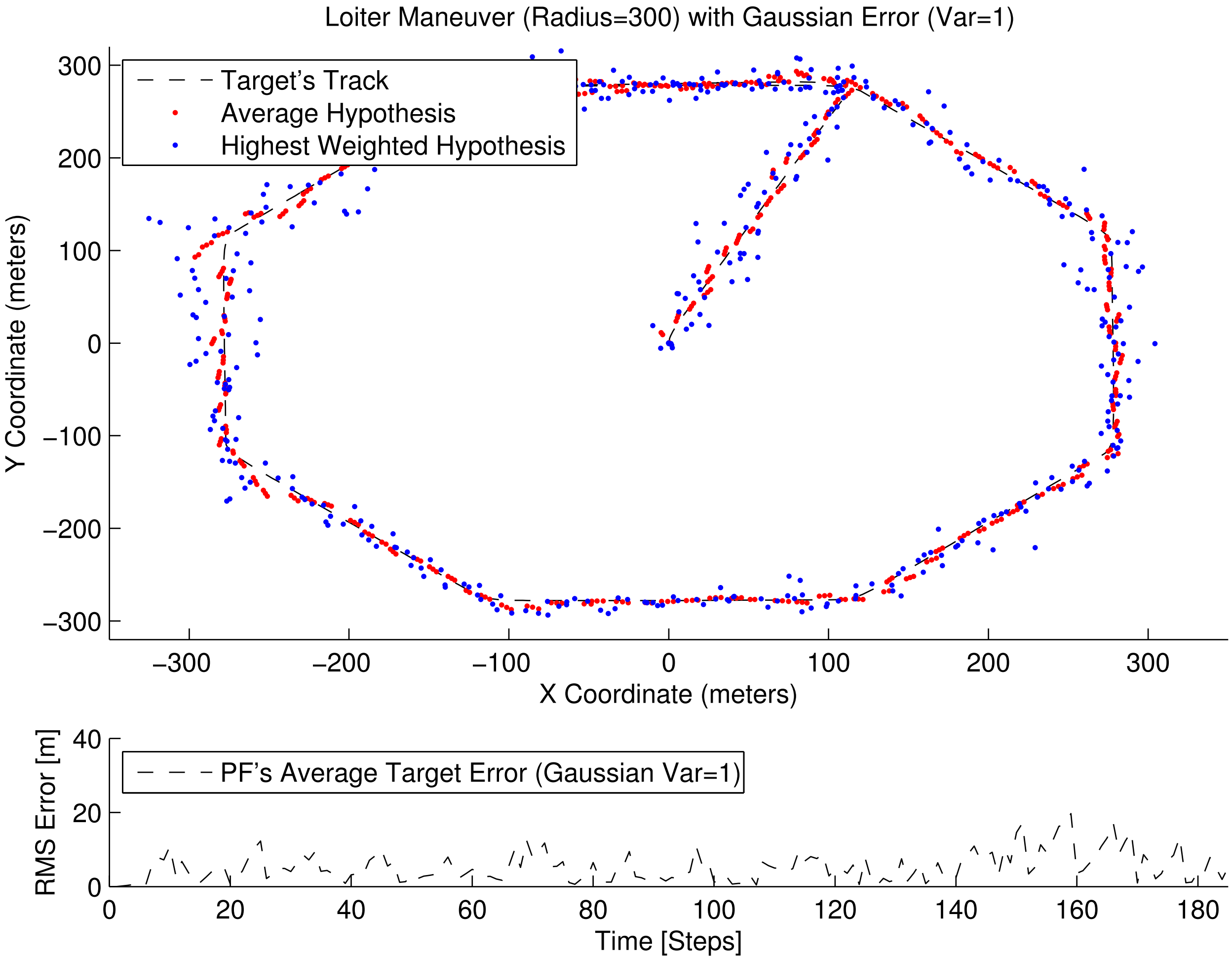


Figure 5-17: Loitering Simulation: Ownship was able to continually track the contact with the particle filter with an approximate error of 10 m. The particle filter tracks with greater accuracy if the contact performs smaller course changes and maintains a longer time on the track segment..

5.3 Collaborative ASV Search

pParticleFilterAP has a collaborative ASV search aspect embedded within. It is based on sharing range information received by both ASV's and using that in their respective particle filters. In the scenario listed (Figure 5-18), both patrol craft were conducting search patterns, once the contact entered within a certain range, patrol craft 2 changed from search mode to pursue. The results of the collaborative particle filter appeared more accurate than the single case. An additional benefit to collaborative search is the reduction in time for the initial localization; this step was rapid because ambiguous solutions were removed as a result of the additional range source. The accuracy improvement was caused from the information of the additional range circle and the increased rate of range sampling caused by having two sensors. Lastly, one patrol craft can share range information with N other patrol craft, given the N patrol craft the benefit of faster localization and accuracy.

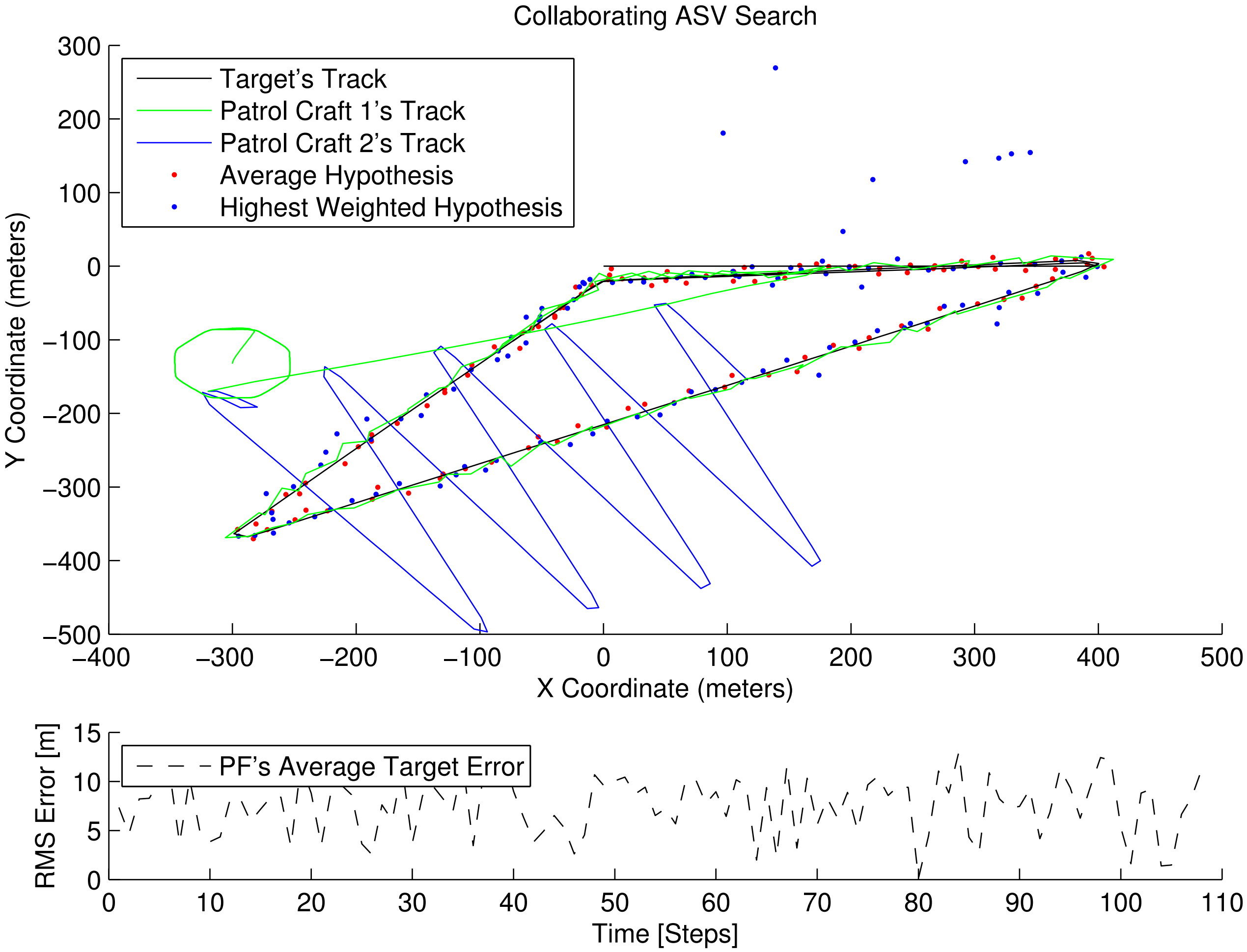


Figure 5-18: Collaborative ASV Search: In this scenario two patrol craft are initially conducting search patterns while sharing range information. As the target travels within 500 meters of patrol craft 2, the patrol craft shifts behavior from a search pattern to pursue the target.

5.4 Charles River Testing

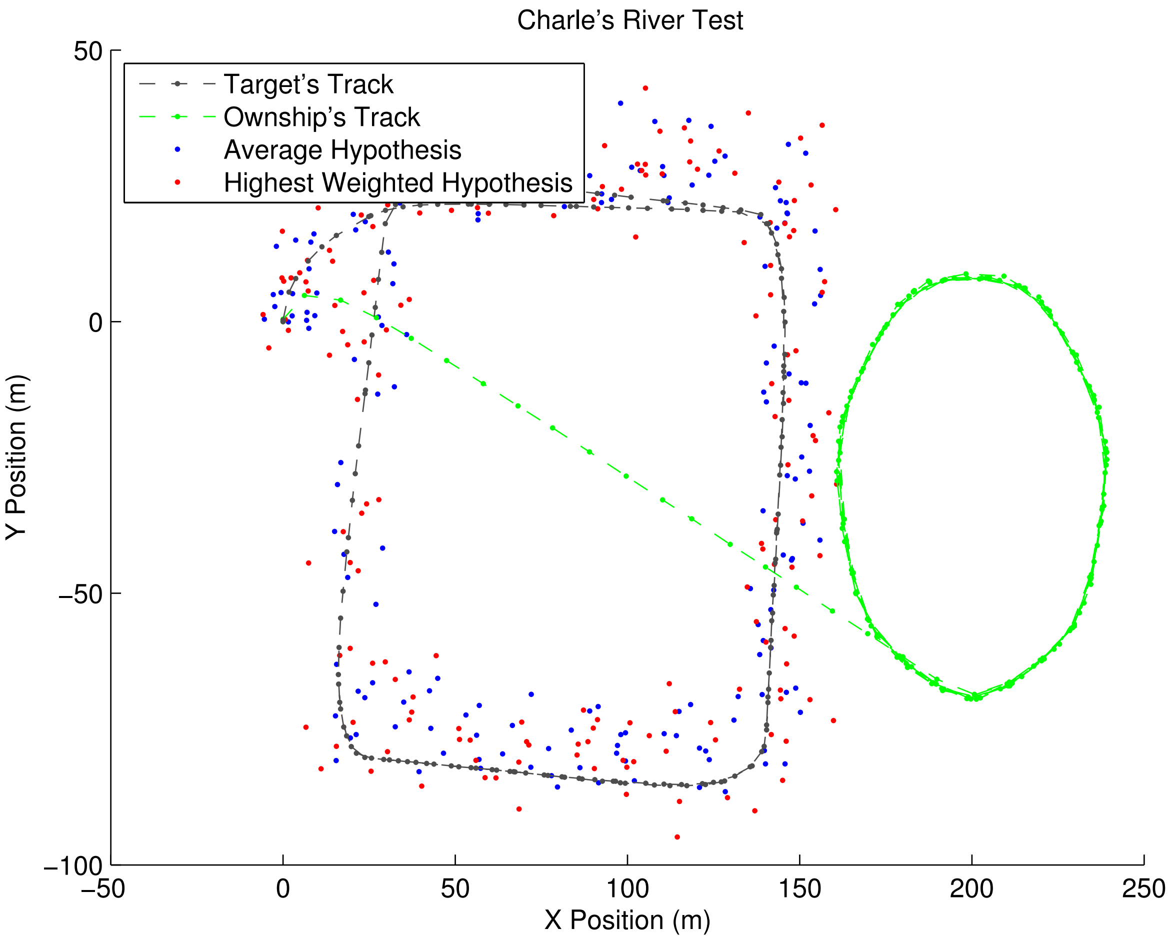


Figure 5-19: Charles River Testing: The particle filter was able to track the contacts within 20 m, note the range measurements used were simulated. The scenario diverged slightly from the previous scenarios by having ownship continually pursue a loiter pattern. This was done to avoid collisions of the vehicle and to protect ownship from pursing a hypothesis located on land.

Due to the limitations of the test environment, the Charles River testing produced mixed results. If the initial state of the contact was known (as mentioned in Section 4.2), the particle filter with the TMA maneuvers were able to track the contact (Figure 5-19). However, the ASV was limited in its ability to localize the contact using only TMA maneuvers. This was a result of the limited WiFi coverage which had a range of 100-200 meters. By the time ownship completed the initialization and TMA maneuvers, the contact had already conducted a course maneuver to stay within the operating envelope. Appendix A contains pictures taken from the testing conducted on the Charles River. The average particle filter error for the maneuvering contact was 14.5 meters for the average weighted hypothesis and 19.6 meters for the highest weighted hypothesis.

5.5 Analysis

One method used to check the results of these simulations and Charles River testing was by comparison of the Cramer-Rao Lower Bound (CLRB) [41]. A CRLB [31] was determine to find the lower bound on the variance of estimators. The bound is derived from the inverse of the Fisher information matrix (FIM). This help determine the expected variance based on the sampling rate and variance of the range measurement. The following equations are a derivation of the CRLB from a Gaussian distribution with a zero mean where $L(X, \sigma^2)$ is the likelihood function [5, 31, 41]. The Fisher Information Matrix J for a single variable is the following:

$$J = -E\left[\frac{\delta^2}{\delta(\sigma^2)^2} \ln L(X, \sigma^2)\right] \tag{5.1}$$

In this case L is the pdf of a Gaussian function with zero mean (μ).

$$L(X, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X-\mu)^2/2\sigma^2} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(X)^2/2\sigma^2} \tag{5.2}$$

$$\ln L(X, \sigma^2) = \frac{\mu}{\sigma^2} X - \frac{1}{2\sigma^2} X^2 - \frac{\mu^2}{\sigma^2} - \ln(\sqrt{2\pi\sigma^2}) \tag{5.3}$$

Since $\mu = 0$, $L(X, \sigma^2)$ is the following:

$$\ln L(X, \sigma^2) = -\frac{1}{2\sigma^2} X^2 - \ln(\sqrt{\sigma^2}) - \ln(\sqrt{2\pi}) \tag{5.4}$$

$$\frac{\delta^2}{\delta(\sigma^2)^2}\left[-\frac{1}{2\sigma^2}X^2 - \ln(\sqrt{\sigma^2}) - \ln(\sqrt{2\pi})\right] \tag{5.5}$$

$$J = -E\left[-\frac{X^2}{4(\sigma^2)^3} + \frac{1}{(\sigma^2)^2}\right] \tag{5.6}$$

$$J = \left[\frac{\sigma^2}{4(\sigma^2)^3} - \frac{1}{(\sigma^2)^2}\right] = \frac{3}{4(\sigma^2)^3} \tag{5.7}$$

Since the CRLB is the inverse of the Fisher Matrix:

$$J^{-1} = \frac{4(\sigma^2)^2}{3} \tag{5.8}$$

And if n independent samples are used:

$$J^{-1} = \frac{4(\sigma^2)^2}{3n} \tag{5.9}$$

The CRLB states that the

$$var(T) =\leq \frac{1}{J} \tag{5.10}$$

so

$$var(T) = \frac{4(\sigma^2)^2}{3n} \tag{5.11}$$

The following figures were constructed to give guidance on the variance of estimators for the sailing pavilion based on the kingfisher ASV. The maximum time steps used was 50, this was based on a max rate of 2 m/s for the Kingfisher with a max WiFi range of 100 m. The results presented in this chapter approached the CRLB estimate with variances of 10 to 15 m.

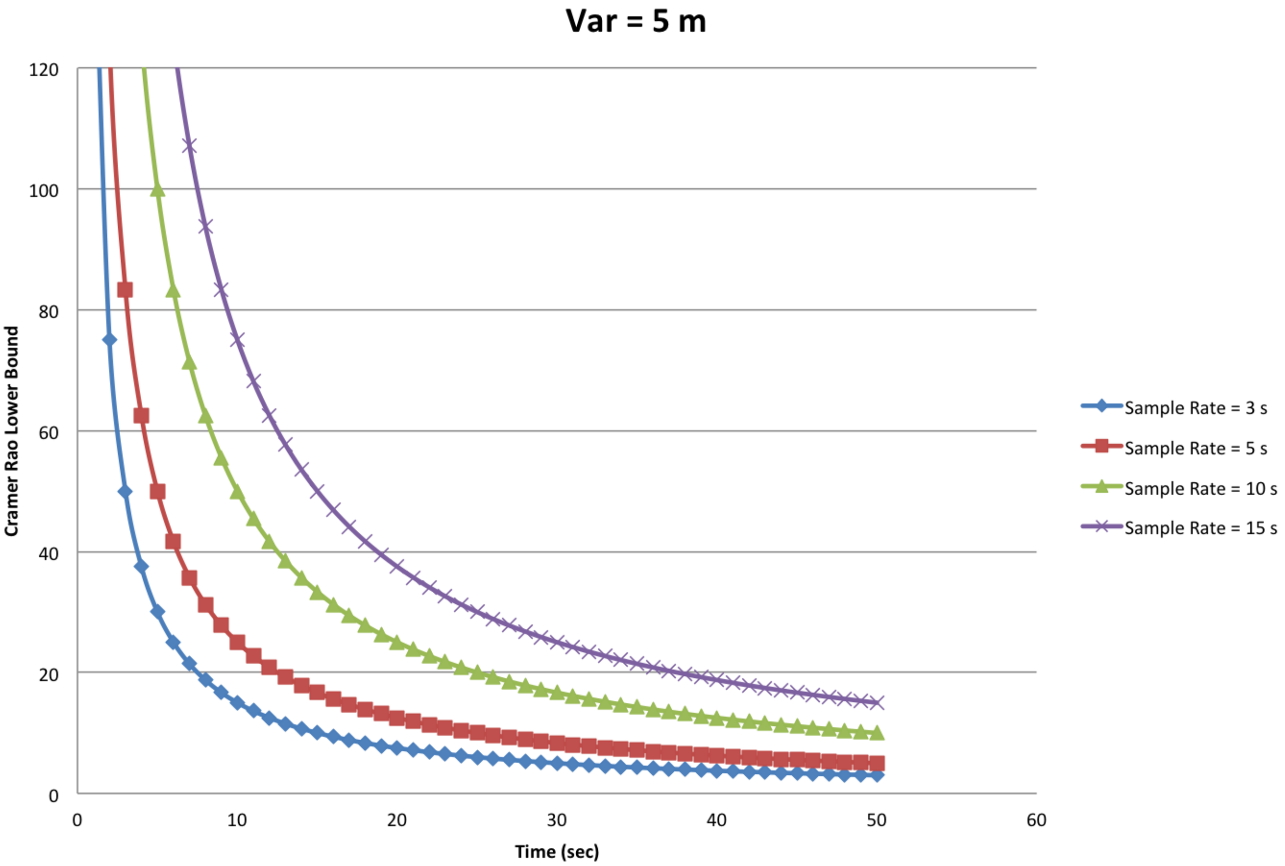


Figure 5-20: CRLB with Variance of 5 meters

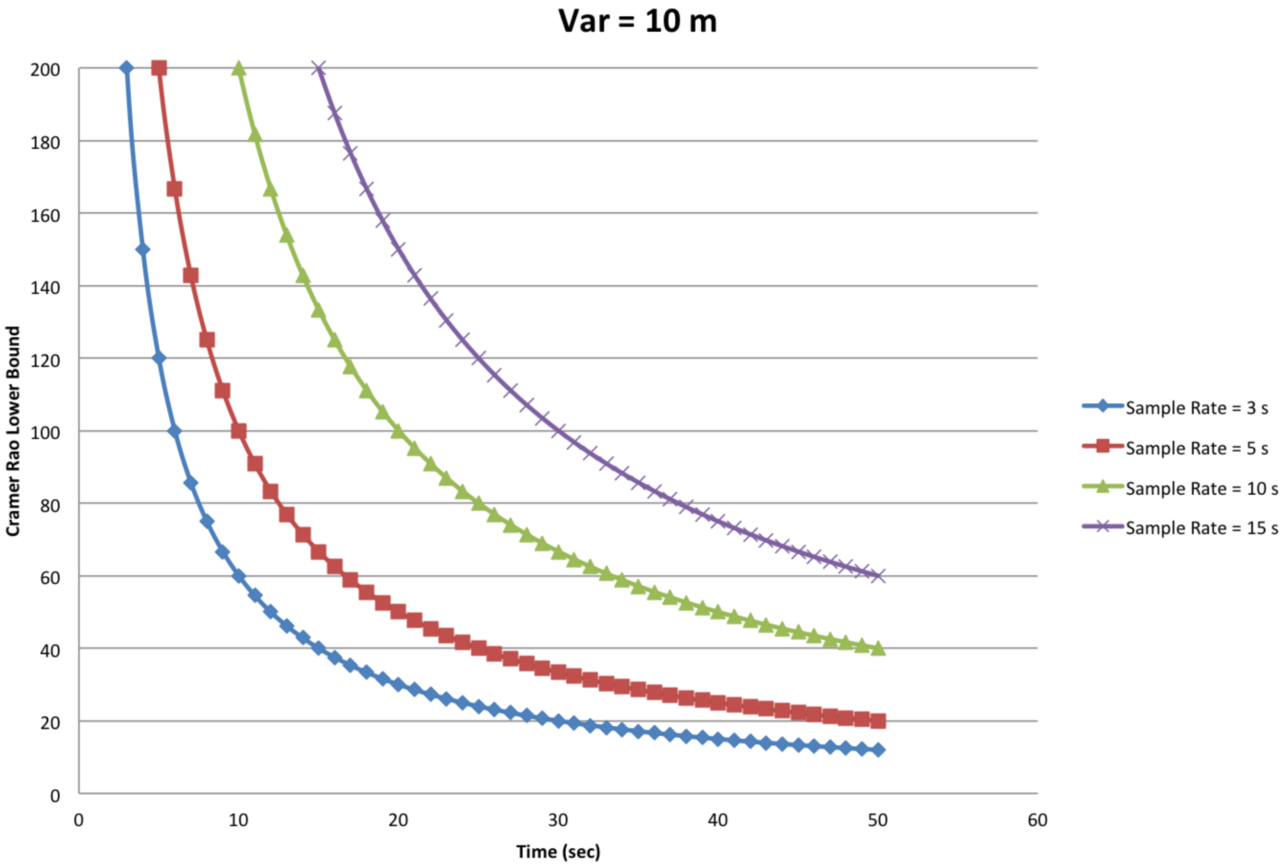


Figure 5-21: CRLB with Variance of 10 meters

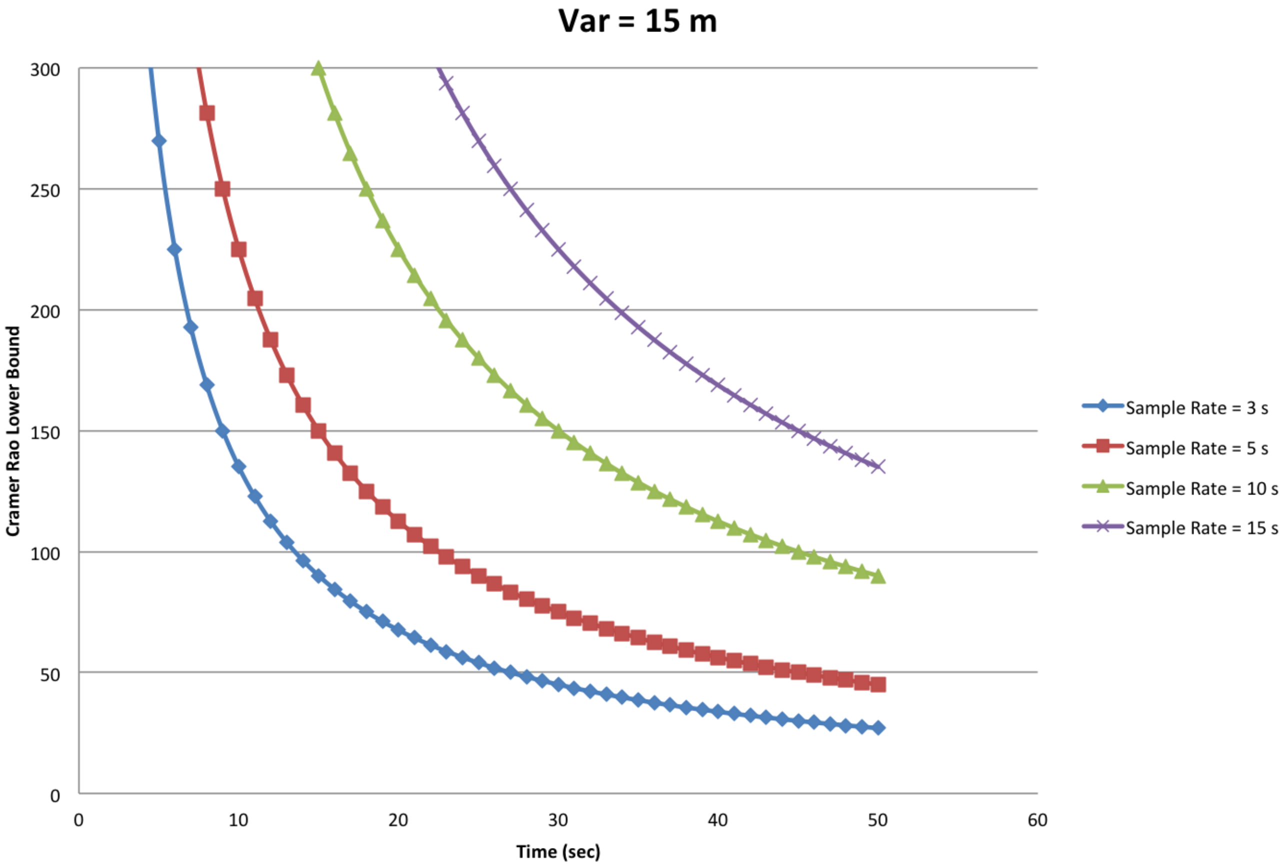


Figure 5-22: CRLB with Variance of 15 meters

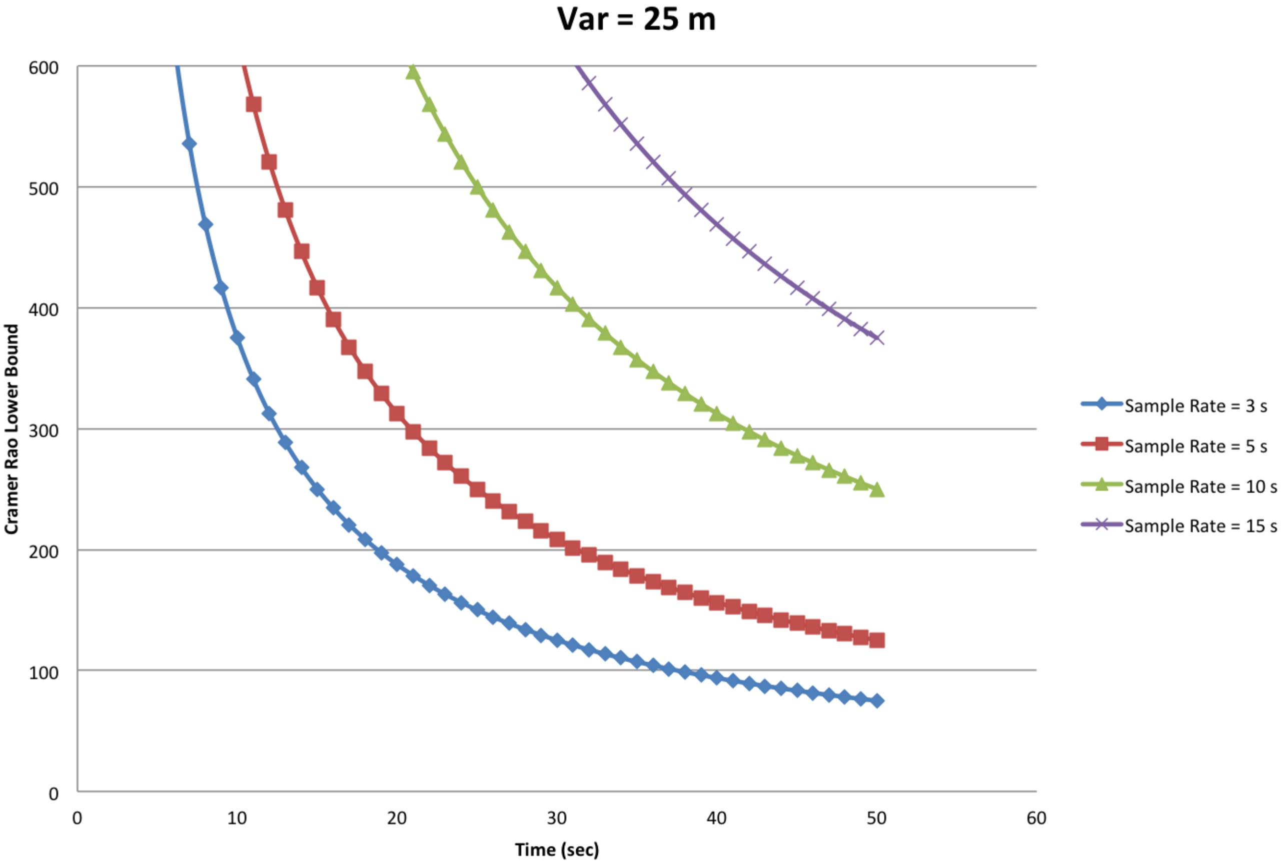


Figure 5-23: CRLB with Variance of 25 meters

Chapter 6

Conclusions

This thesis was able to show that a particle filter coupled with TMA based maneuvers could use range-only data to localize and track a single contact through collaborating with multiple vehicles or conducting adaptive maneuvering that removed particle ambiguity. While tracking stationary and moving contacts appeared to be reasonable for the particle filter, tracking maneuvering contacts proved to be more difficult especially as the angle and frequency of the maneuvers increased. Reserve particles were used to aid in the tracking of maneuvering contacts and they proved to be a viable solution. The particle filter was able to track a single moving contact within 5 meters of accuracy and a maneuvering contact within15 meters.

It is important to recall the limits of this method as noted from the lack of observability in the state equation as discussed in Chapter 4 and the CRLB discussed in Chapter 5.

6.1 Future Work

There are many areas in which pParticleFilterAP could be improved and researched.

1. The first involves the pdf that was used to determine the weights. This could be investigated and optimized. One such way to improve it would be to incorporate local land and interference information and discard those possibilities from the probability distribution those courses and positions that are not feasible.

2. Deeper exploration of the particle filter parameters of noise, Gaussian variance, and number of particles. Some of these factors were investigated, it was done so empirically. Finding an understanding of why and how to set the values would be useful.

3. The particle filter that was used in this thesis was calibrated from the previously mentioned parameters. Finding an autonomous way for calibrating would be ideal for adaptability to numerous environments.

4. The use of different vehicles, such as the REMUS AUV [38], could be used to conduct underwater experiments and verifications.

5. It seems necessary to perform additional testing with multiple vehicles, as to, incorporate the capability for the vehicles to maneuver to optimize the performance of the tracker. In our results in the previous chapters, the vehicles shared raw range data to obtain a better particle filter solution. A desirable next step is to compute cooperative maneuvers for the vehicles in order to minimize the uncertainty in the target location, as discussed in [3, 43].

6. Further research could be to build and compare more filters such as a Maximum Likelihood Estimator (MLE) or an Extended Kalman Filter (EKF).

7. Implement a method to track multiple contacts; one of the assumptions that was made early on was that there was only one target. [13]

8. There is a need to expand the WiFi Coverage of the Charles River to increase the range of the ASVs and the physical domain for the testing.

9. More investigation into the state equations used for observability is needed and the development of additional behaviors to maneuver to overcome ambiguity would be useful.

10. It may prove fruitful to develop an interacting multiple modeling filter approach for tracking, and to integrated it with the IvPHelm.

11. Finally, even though the project focused on processing sonar information for submerged contacts, the application developed should be capable of handling radar range measurements. It would be interesting to adapt the MOOS-IvP modules developed in this thesis for radar tracking of surface contacts.

Appendix A

Charles River Testing Pictures

Figure A-1: Kingfisher at sea testing

Figure A-2: Kingfisher at sea 2

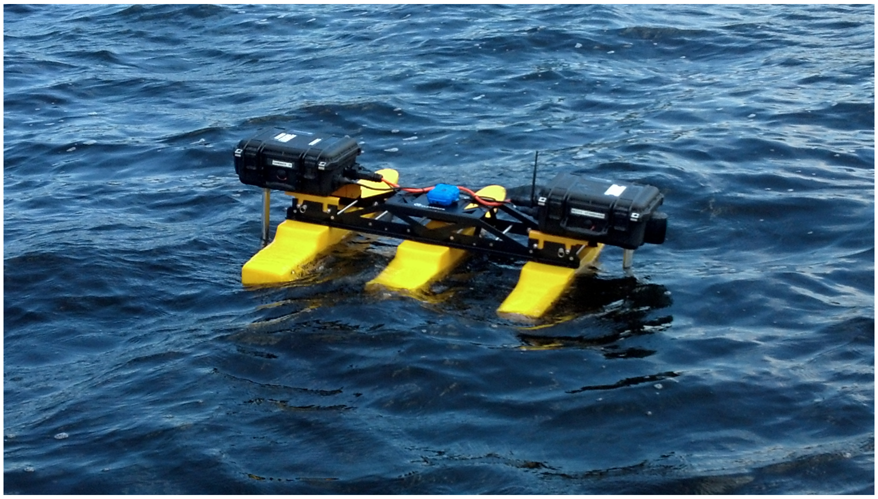


Figure A-3: Kingfisher at sea 3

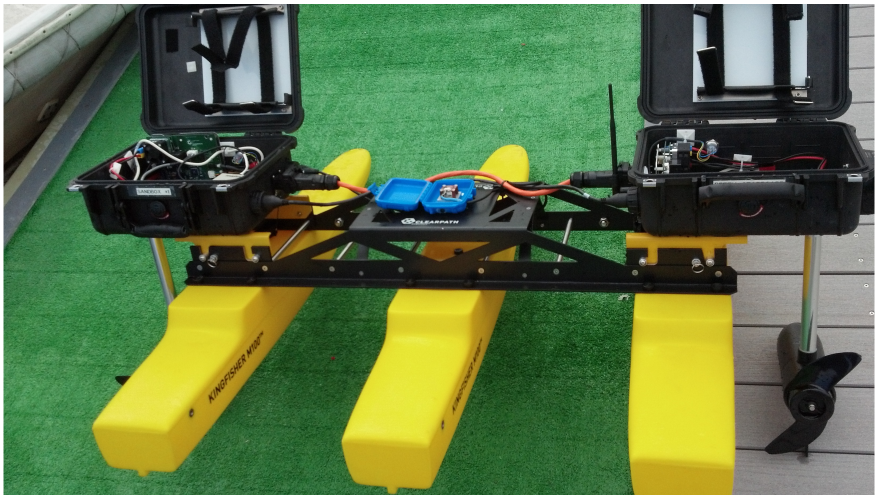


Figure A-4: Kingfisher internal 1

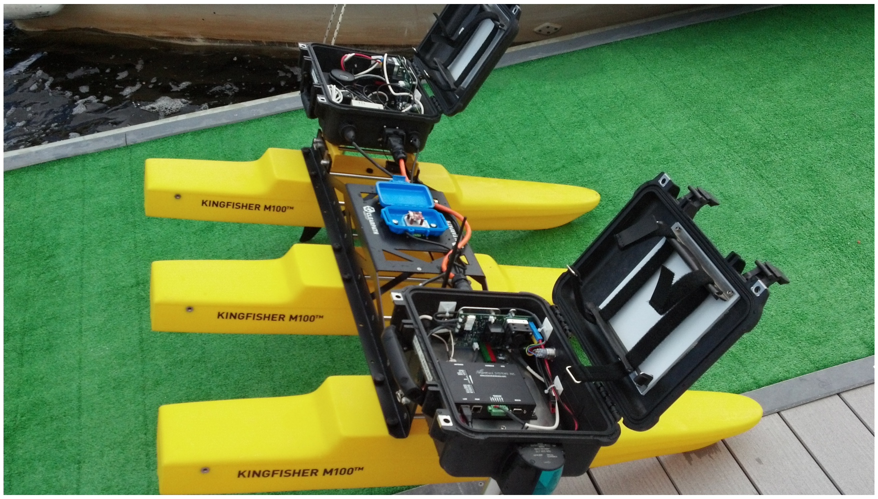


Figure A-5: Kingfisher internal 2

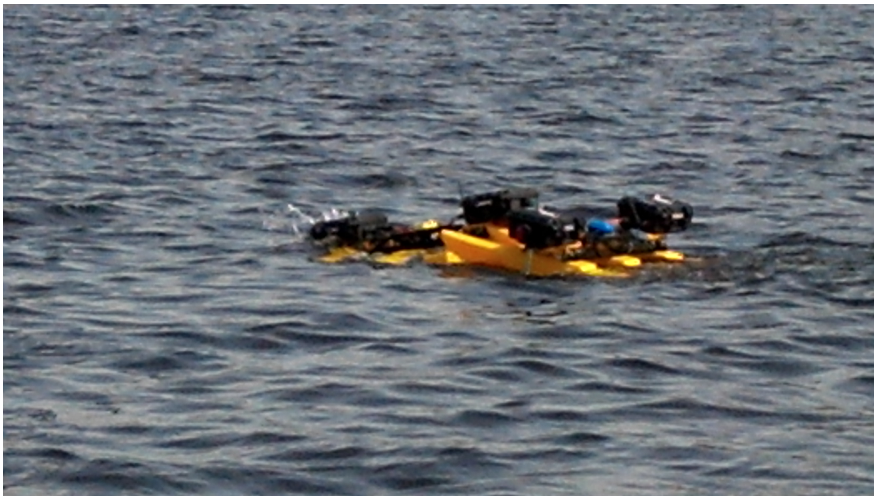


Figure A-6: Up Close Tracking: In this picture ownship collided with the contact. This is an example of how well the tracking algorithm works; however, a safety method to prevent collisions needs to be addressed.

Bibliography

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–187, February 2002.

[2] A. B. Baggeroer. Acoustic telemetry — an overview. *IEEE Journal of Oceanic Engineering*, 9(4):229–235, October 1984.

[3] A. Bahr. *Cooperative Localization for Autonomous Underwater Vehicles.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, February 2009.

[4] A. Bahr, J.J. Leonard, and M.F. Fallon. Cooperative localization for autonomous underwater vehicles. *Intl. J. of Robotics Research*, 28(6):714–728, 2009.

[5] Y. Bar-Shalom and Xiao-Rong Li. *Estimation and Tracking: Principles, Techniques, and Software.* Artech House, Inc., 1993.

[6] Janice C. Beaver. U.S. International Borders: Brief Facts, 2009.

[7] M. R. Benjamin. *Interval Programming: A Multi-Objective Optimiza- tion Model for Autonomous Vehicle Control.* PhD thesis, Brown University, May 2002.

[8] Michael R. Benjamin. *Interval Programming: A Multi-Objective Optimization Model for Autonomous Vehicle Control.* PhD thesis, Brown University, 2002.

[9] Michael R. Benjamin. The moos-ivp ufield toolbox for multi-vehicle operations and simulation. Technical report, MIT, 2012.

[10] M.R. Benjamin. Multi-objective autonomous vehicle navigation in the presence of cooperative and adversarial moving contacts. In *OCEANS'02 MTS/IEEE*, volume 3, pages 1878–1885. IEEE, 2002.

[11] M.R. Benjamin, J.J. Leonard, H. Schmidt, and P. Newman. An overview of MOOS-IvP and a users guide to the IvP helm autonomy software - release 12.2. http://oceanai.mit.edu/moos-ivp-pdf/moosivp-helm.pdf, January 2012.

[12] Samuel Blackman and Robert Popoli. *Design and Analysis of Modern Tracking Systems.* Artech House, Inc., 1999.

[13] V. Cevher, R. Velmurugan, and J.H. McClellan. A range-only multiple target particle filter tracker. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 4, pages IV–IV. IEEE, 2006.

[14] Andrew Cunningham and Bruce Thomas. Target motion analysis visualisation. *Conferences in Research and Practice in Information Technology*, 45, 2005.

[15] Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Monte Carlo localization for mobile robots. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 1999.

[16] M.F. Fallon, G. Papadopoulos, and J.J. Leonard. Cooperative AUV navigation using a single surface craft. In *Field and Service Robotics (FSR)*, July 2009.

[17] M. H. Ferdowsi. Observability conditions for target states with bearing-only measurements in three-dimensional case. *International Conference on Control Applications*, 2006.

[18] J. C. Harsfield. Single transponder range only navigation geometry (strong) applied to remus autonomous under water vehicles. Master's thesis, Massachusetts Institute of Technology, 2005.

[19] M. Isard. *Visual Motion Analysis by Probabilistic Propagation of Conditional Density.* PhD thesis, Oxford University, 1998.

[20] D. B. Kilfoyle and A.B. Baggeroer. The state of the art in underwater acoustic telemetry. *IEEE Journal of Oceanic Engineering*, 25(1):4–27, January 2000.

[21] J.C. Kinsey, R.M. Eustice, and L.L. Whitcomb. A survey of underwater vehicle navigation: Recent advances and new challenges. In *IFAC Conference of Manoeuvering and Control of Marine Craft*, 2006.

[22] J. J. Leonard, A. A. Bennett, C. M. Smith, and H. J. S. Feder. Autonomous underwater vehicle navigation. Technical Report Marine Robotics Laboratory Technical Memorandum 98-1, MIT, 1998.

[23] Magean La Mala. The narco submarine. *Vivirlatino*, 2008.

[24] Amy McCullough. Legislation targets drug smuggling subs. *Navy Times*, 2008.

[25] J.S. Milton and J.C. Arnold. *Introduction to probability and statistics: principles and applications for engineering and the computing sciences.* McGraw-Hill, Inc., 2002.

[26] S. C. Nardone and V. J. Aidala. Observability Criteria For Bearings-Only Target Motion Analysis. *IEEE Transactions on Aerospace and Electronic Systems*, 17(2):162–166, 1981.

[27] P.M. Newman. MOOS - a mission oriented operating suite. Technical Report OE2003-07, Department of Ocean Engineering, MIT, 2003.

[28] American Association of Port Authorities. U.S. Public Port Facts. http://www.aapa-ports.org/files/PDFs/facts.pdf, 2012 April.

[29] U.S. Department of Transportation Federal Highway Administration. Freight facts and figures 200, July 2009.

[30] S. Petillo, H. Schmidt, and A. Balasuriya. Constructing a distributed auv network for underwater plume-tracking operations. *International Journal of Distributed Sensor Networks*, 2012(191235):12, 2012.

[31] B. Ristic, S. Arulampalam, and J. McCarthy. Target motion analysis using range-only measurements: Algorithms, performance, and application to Ingara ISAR data. Technical Report DSTO-TR-1095, Defence Science and Technology Organisation, 2001.

[32] Clearpath Robotics. Clearpath Robotics. www.clearpathrobotics.com, April 2012.

[33] M. Shafer, Andrew J. Autonomous cooperation of heterogeneous platforms for sea-based search tasks. Master's thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science., 2008.

[34] Taek Lyul Song. Observability of target tracking with range-only measurements. *IEEE Journal of Oceanic Engineering*, 24(3):383–387, July 1999.

[35] T.Brehard and J. P. Le Cadre. Hierarchical particle filter for bearings-only trackin. *IEEE Transactions on Aerospace and Electronic Systems*, 42(4):1567–1585, October 2007.

[36] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* The MIT press, Cambridge, MA, 2005.

[37] Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128, May 2001.

[38] C. von Alt, B. Allen, T. Austin, and R. Stokey. Remote environmental monitoring units. In *AUV 94*, 1994.

[39] D. H. Wagner, W. C. Mylander, and T. J. Sanders. *Naval Operations Analysis.* Naval Institute Press, 1999.

[40] Alan R. Washburn. Random Meetings at Sea. Technical Report NPS-OR-10-001, Naval Post Graduate School, 2010.

[41] E. J. Wegman and J. C. Smithh. *Statistical Signal Processing.* Marcel Dekker Inc, 1984.

[42] K. Zhou and S. I. Roumeliotis. Optimal motion strategies for range-only optimal motion strategies for distributed target tracking. Technical Report 2006-004, University of Minnesota, 2006.

[43] K. Zhou and S.I. Roumeliotis. Optimal motion strategies for range-only constrained multisensor target tracking. *Robotics, IEEE Transactions on*, 24(5):1168–1185, 2008.